

Average-Time Games on Timed Automata

Marcin Jurdziński¹, and Ashutosh Trivedi²

¹ Department of Computer Science, University of Warwick, UK

² Computing Laboratory, University of Oxford, UK.

Abstract. An average-time game is played on the infinite graph of configurations of a finite timed automaton. The two players, Min and Max, construct an infinite run of the automaton by taking turns to perform a timed transition. Player Min wants to minimise the average time per transition and player Max wants to maximise it. A solution of average-time games is presented using a reduction to average-price game on a finite graph. A direct consequence is an elementary proof of determinacy for average-time games. This complements our results for reachability-time games and partially solves a problem posed by Bouyer et al., to design an algorithm for solving average-price games on priced timed automata. The paper also establishes the exact computational complexity of solving average-time games: the problem is EXPTIME-complete for timed automata with at least two clocks.

1 Introduction

Real-time open systems are computational systems that interact with environment and whose correctness depends critically on the time at which they perform some of their actions. The problem of design and verification of such systems can be formulated as *two-player zero-sum games*. A heart pacemaker is an example of a real-time open system as it interacts with the environment (heart, body movements, and breathing) and its correctness depends critically on the time at which it performs some of its actions (sending pace signals to the heart in real time). Other examples of safety-critical real-time open systems include nuclear reactor protective systems, industrial process controllers, aircraft-landing scheduling systems, satellite-launching systems, etc. Designing correct real-time systems is of paramount importance. Timed automata [2] are a popular and well-established formalism for modelling real-time systems, and games on timed automata can be used to model real-time open systems. In this paper, we introduce *average-time games* which model the interaction between the real-time open system and the environment; and we are interested in finding a strategy of the system which results in minimum average-time per transition, assuming adversarial environment.

Related Work. Games with quantitative payoffs can be studied as a model for optimal-controller synthesis [3,1,6]. Among various quantitative payoffs the average-price payoff [9,8] is the most well-studied in game theory, Markov decision processes, and planning literature [8,16], and it has numerous appealing

interpretations in applications. Most algorithms for solving Markov decision processes [16] or games with average-price payoff work for finite graphs only [18,8]. Asarin and Maler [3] presented the first algorithm for games on timed automata (timed games) with a quantitative payoff: reachability-time payoff. Their work was later generalised by Alur et al. [1] and Bouyer et al. [6] to give partial decidability results for reachability-price games on linearly-priced timed automata. The exact computational complexity of deciding the value in timed games with reachability-time payoff was shown to be EXPTIME in [11,7]. Bouyer et al. [5] also studied the more difficult average-price payoffs, but only in the context of scheduling, which in game-theoretic terminology corresponds to 1-player games. They left open the problem of proving decidability of 2-player average-reward games on linearly-priced timed automata. We have recently extended the results of Bouyer et al. to solve 1-player games on more general concavely-priced timed automata [12]. In this paper we address the important and non-trivial special case of average-time games (i.e., all locations have unit costs), which was also left open by Bouyer et al.

Our Contributions. Average-time games on timed automata are introduced. This paper gives an elementary proof of determinacy for these games. A new type of region [2] based abstraction—boundary region graph—is defined, which generalises the corner-point abstraction of Bouyer et al. [5]. Our solution allows computing the value of average-time games for an arbitrary starting state (i.e., including non-corner states). Finally, we establish the exact complexity of solving average-time games: the problem is EXPTIME-complete for timed automata with at least two clocks.

Organisation of the Paper. In Section 2 we discuss average-price games (also known as mean-payoff games) on finite graphs and cite some important results for these games. In Section 3 we introduce average-time games on timed automata. In Section 4 we introduce some region-based abstractions of timed automata, including the closed region graph, and its subgraphs: the boundary region graph, and the region graph. While the region graph is semantically equivalent to the corresponding timed automaton, the boundary region graph has the property that for every starting state, the reachable state space is finite. We introduce average-time games on these graphs and in Section 6 we show that if we have the solution of the average-time game for any of these graphs, then we get the solution of the average-time game for the corresponding timed automaton. Finally, in Section 7 we discuss the computational complexity of solving average-time games.

Notations. We assume that, wherever appropriate, sets \mathbb{Z} of integers, \mathbb{N} of non-negative integers and \mathbb{R} of reals contain a maximum element ∞ , and we write \mathbb{N}_+ for the set of positive integers and \mathbb{R}_\oplus for the set of non-negative reals. For $n \in \mathbb{N}$, we write $\langle n \rangle_{\mathbb{N}}$ for the set $\{0, 1, \dots, n\}$, and $\langle n \rangle_{\mathbb{R}}$ for the set $\{r \in \mathbb{R} : 0 \leq r \leq n\}$ of non-negative reals bounded by n . For a real number $r \in \mathbb{R}$, we write $|r|$ for its absolute value, we write $\lfloor r \rfloor$ for its integer part, i.e., the largest integer $n \in \mathbb{N}$, such that $n \leq r$, and we write $\{r\}$ for its fractional part, i.e., we have $\{r\} = r - \lfloor r \rfloor$.

2 Average-Price Games

A (perfect-information) two-player *average-price game* [18,8] (also known as mean-payoff game) $\Gamma = (V, E, V_{\text{Max}}, V_{\text{Min}}, p)$ consists of a finite directed graph (V, E) , a partition $V = V_{\text{Max}} \cup V_{\text{Min}}$ of vertices, and a *price function* $\pi : E \rightarrow \mathbb{Z}$. A play starts at a vertex $v_0 \in V$. If $v_0 \in V_p$, for $p \in \{\text{Max}, \text{Min}\}$, then player p chooses a successor of the current vertex v_0 , i.e., a vertex v_1 , such that $(v_0, v_1) \in E$, and v_1 becomes the new current vertex. When this happens then we say that player p has made a move from the current vertex. Players keep making moves in this way indefinitely, thus forming an infinite path $r = (v_0, v_1, v_2, \dots)$ in the game graph. The goal of player Min is to minimise $\mathcal{A}_{\text{Min}}(r) = \limsup_{n \rightarrow \infty} (1/n) \cdot \sum_{i=1}^n \pi(v_{i-1}, v_i)$ and the goal of player Max is to maximise $\mathcal{A}_{\text{Max}}(r) = \liminf_{n \rightarrow \infty} (1/n) \cdot \sum_{i=1}^n \pi(v_{i-1}, v_i)$.

Strategies for players are defined as usual [18,8]. We write Σ_{Min} (Σ_{Max}) for the set of strategies of player Min (Max) and Π_{Min} (Π_{Max}) for the set of positional strategies of player Min (Max). For strategies $\mu \in \Sigma_{\text{Min}}$ and $\chi \in \Sigma_{\text{Max}}$, and for an initial vertex $v \in V$, we write $\text{run}(v, \mu, \chi)$ for the unique path formed if players start in the vertex v and then they follow strategies μ and χ , respectively. For brevity, we write $\mathcal{A}_{\text{Min}}(v, \mu, \chi)$ for $\mathcal{A}_{\text{Min}}(\text{run}(v, \mu, \chi))$ and we write $\mathcal{A}_{\text{Max}}(v, \mu, \chi)$ for $\mathcal{A}_{\text{Max}}(\text{run}(v, \mu, \chi))$.

For a vertex $v \in V$, we define the *upper value* as

$$\overline{\text{val}}(v) = \inf_{\mu \in \Sigma_{\text{Min}}} \sup_{\chi \in \Sigma_{\text{Max}}} \mathcal{A}_{\text{Min}}(v, \mu, \chi),$$

and the *lower value* as

$$\underline{\text{val}}(v) = \sup_{\chi \in \Sigma_{\text{Max}}} \inf_{\mu \in \Sigma_{\text{Min}}} \mathcal{A}_{\text{Max}}(v, \mu, \chi).$$

Note that the inequality $\underline{\text{val}}(v) \leq \overline{\text{val}}(v)$ always holds. A game is determined if for every $v \in V$, we have $\underline{\text{val}}(v) = \overline{\text{val}}(v)$. We then write $\text{val}(v)$ for this number and we call it the *value* of the average-price game at the vertex v .

We say that the strategies $\mu^* \in \Sigma_{\text{Min}}$ and $\chi^* \in \Sigma_{\text{Max}}$ are *optimal* for the respective players, if for every vertex $v \in V$, we have that $\sup_{\chi \in \Sigma_{\text{Max}}} \mathcal{A}_{\text{Min}}(v, \mu^*, \chi) = \overline{\text{val}}(v)$ and $\inf_{\mu \in \Sigma_{\text{Min}}} \mathcal{A}_{\text{Max}}(v, \mu, \chi^*) = \underline{\text{val}}(v)$. Liggett and Lippman [13] show that all perfect-information (stochastic) average-price games are positionally determined.

Theorem 1. [13] *Every average-price game is determined, and optimal positional strategies exist for both players, i.e., for all $v \in V$, we have:*

$$\inf_{\mu \in \Pi_{\text{Min}}} \sup_{\chi \in \Sigma_{\text{Max}}} \mathcal{A}_{\text{Min}}(v, \mu, \chi) = \sup_{\chi \in \Pi_{\text{Max}}} \inf_{\mu \in \Sigma_{\text{Min}}} \mathcal{A}_{\text{Max}}(v, \mu, \chi).$$

The decision problem for average-price games is in $\text{NP} \cap \text{co-NP}$; no polynomial-time algorithm is currently known for the problem.

3 Average-Time Games

3.1 Timed Automata

Before we present the syntax of the timed automata, we need to introduce some concepts. Fix a constant $k \in \mathbb{N}$ for the rest of this paper. Let C be a finite set of *clocks*. Clocks in timed automata are usually allowed to take arbitrary non-negative real values. For the sake of simplicity and w.l.o.g [4], we restrict them to be bounded by some constant k , i.e., we consider only *bounded* timed automata models. A (k -bounded) *clock valuation* is a function $\nu : C \rightarrow \llbracket k \rrbracket_{\mathbb{R}}$; we write \mathcal{V} for the set $[C \rightarrow \llbracket k \rrbracket_{\mathbb{R}}]$ of clock valuations. If $\nu \in \mathcal{V}$ and $t \in \mathbb{R}_{\oplus}$ then we write $\nu + t$ for the clock valuation defined by $(\nu + t)(c) = \nu(c) + t$, for all $c \in C$. For a set $C' \subseteq C$ of clocks and a clock valuation $\nu : C \rightarrow \mathbb{R}_{\oplus}$, we define $\text{reset}(\nu, C')(c) = 0$ if $c \in C'$, and $\text{reset}(\nu, C')(c) = \nu(c)$ if $c \notin C'$. A *corner* is an integer clock valuation, i.e., α is a corner if $\alpha(c) \in \llbracket k \rrbracket_{\mathbb{N}}$, for every clock $c \in C$.

The set of *clock constraints* over the set of clocks C is the set of conjunctions of *simple clock constraints*, which are constraints of the form $c \bowtie i$ or $c - c' \bowtie i$, where $c, c' \in C$, $i \in \llbracket k \rrbracket_{\mathbb{N}}$, and $\bowtie \in \{<, >, =, \leq, \geq\}$. There are finitely many simple clock constraints. For every clock valuation $\nu \in \mathcal{V}$, let $\text{SCC}(\nu)$ be the set of simple clock constraints which hold in $\nu \in \mathcal{V}$. A *clock region* is a maximal set $P \subseteq \mathcal{V}$, such that for all $\nu, \nu' \in P$, $\text{SCC}(\nu) = \text{SCC}(\nu')$. In other words, every clock region is an equivalence class of the indistinguishability-by-clock-constraints relation, and vice versa. Note that ν and ν' are in the same clock region iff all clocks have the same integer parts in ν and ν' , and if the partial orders of the clocks, determined by their fractional parts in ν and ν' , are the same. For all $\nu \in \mathcal{V}$, we write $[\nu]$ for the clock region of ν . A *clock zone* is a convex set of clock valuations, which is a union of a set of clock regions. Note that a set of clock valuations is a zone iff it is definable by a clock constraint. For $W \subseteq \mathcal{V}$, we write $\text{clos}(W)$ for the smallest closed set in \mathcal{V} which contains W . Observe that for every clock zone W , the set $\text{clos}(W)$ is also a clock zone.

Let L be a finite set of *locations*. A *configuration* is a pair (ℓ, ν) , where $\ell \in L$ is a location and $\nu \in \mathcal{V}$ is a clock valuation; we write Q for the set of configurations. If $s = (\ell, \nu) \in Q$ and $c \in C$, then we write $s(c)$ for $\nu(c)$. A *region* is a pair (ℓ, P) , where ℓ is a location and P is a clock region. If $s = (\ell, \nu)$ is a configuration then we write $[s]$ for the region $(\ell, [\nu])$. We write \mathcal{R} for the set of regions. A set $Z \subseteq Q$ is a *zone* if for every $\ell \in L$, there is a clock zone W_{ℓ} (possibly empty), such that $Z = \{(\ell, \nu) : \ell \in L \text{ and } \nu \in W_{\ell}\}$. For a region $R = (\ell, P) \in \mathcal{R}$, we write $\text{clos}(R)$ for the zone $\{(\ell, \nu) : \nu \in \text{clos}(P)\}$.

A *timed automaton* $\mathcal{T} = (L, C, S, A, E, \delta, \varrho)$ consists of a finite set of locations L , a finite set of clocks C , a set of *states* $S \subseteq Q$, a finite set of *actions* A , an *action enabledness function* $E : A \rightarrow 2^S$, a *transition function* $\delta : L \times A \rightarrow L$, and a *clock reset function* $\varrho : A \rightarrow 2^C$. We require that S , and $E(a)$ for all $a \in A$, are zones.

Clock zones, from which zones S , and $E(a)$, for all $a \in A$, are built, are typically specified by clock constraints. Therefore, when we consider a timed automaton as an input of an algorithm, its size should be understood as the sum

of sizes of encodings of L , C , A , δ , and ϱ , and the sizes of encodings of clock constraints defining zones S , and $E(a)$, for all $a \in A$. Our definition of a timed automaton may appear to differ from the usual ones [2,4], but the differences are superficial.

For a configuration $s = (\ell, \nu) \in Q$ and $t \in \mathbb{R}_\oplus$, we define $s + t$ to be the configuration $s' = (\ell, \nu + t)$ if $\nu + t \in \mathcal{V}$, and we then write $s \rightarrow_t s'$. We write $s \rightarrow_t s'$ if $s \rightarrow_t s'$ and for all $t' \in [0, t]$, we have $(\ell, \nu + t') \in S$. For an action $a \in A$, we define $\text{succ}(s, a)$ to be the configuration $s' = (\ell', \nu')$, where $\ell' = \delta(\ell, a)$ and $\nu' = \text{reset}(\nu, \varrho(a))$, and we then write $s \xrightarrow{a} s'$. We write $s \xrightarrow{a} s'$ if $s \xrightarrow{a} s'$; $s, s' \in S$; and $s \in E(a)$. For technical convenience, and without loss of generality, we will assume throughout that for every $s \in S$, there exists $a \in A$, such that $s \xrightarrow{a} s'$. For $s, s' \in S$, we say that s' is in the future of s , or equivalently, that s is in the past of s' , if there is $t \in \mathbb{R}_\oplus$, such that $s \rightarrow_t s'$; we then write $s \rightarrow_* s'$.

For $R, R' \in \mathcal{R}$, we say that R' is in the future of R , or that R is in the past of R' , if for all $s \in R$, there is $s' \in R'$, such that s' is in the future of s ; we then write $R \rightarrow_* R'$. Similarly, for $R, R' \in \mathcal{R}$, we write $R \xrightarrow{a} R'$ if there is $s \in R$, and there is $s' \in R'$, such that $s \xrightarrow{a} s'$.

A *timed action* is a pair $\tau = (t, a) \in \mathbb{R}_\oplus \times A$. For $s \in Q$, we define $\text{succ}(s, \tau) = \text{succ}(s, (t, a))$ to be the configuration $s' = \text{succ}(s + t, a)$, i.e., such that $s \rightarrow_t s'' \xrightarrow{a} s'$, and we then write $s \xrightarrow{\tau} s'$. We write $s \xrightarrow{\tau} s'$ if $s \rightarrow_t s'' \xrightarrow{a} s'$, and we then say that $(s, (t, a), s')$ is a *transition* of the timed automaton. If $\tau = (t, a)$ then we write $s \xrightarrow{\tau} s'$ instead of $s \xrightarrow{a} s'$, and $s \xrightarrow{\tau} s'$ instead of $s \xrightarrow{a} s'$.

An infinite run of a timed automaton is a sequence $r = \langle s_0, \tau_1, s_1, \tau_2, \dots \rangle$, such that for all $i \geq 1$, we have $s_{i-1} \xrightarrow{\tau_i} s_i$. A finite run of a timed automaton is a finite sequence $\langle s_0, \tau_1, s_1, \tau_2, \dots, \tau_n, s_n \rangle \in S \times ((A \times \mathbb{R}_\oplus) \times S)^*$, such that for all i , $1 \leq i \leq n$, we have $s_{i-1} \xrightarrow{\tau_i} s_i$. For a finite run $r = \langle s_0, \tau_1, s_1, \tau_2, \dots, \tau_n, s_n \rangle$, we define $\text{length}(r) = n$, and we define $\text{last}(r) = s_n$ to be the state in which the run ends. For a finite run $r = \langle s_0, \tau_1, s_1, \tau_2, \dots, s_n \rangle$, we define time of the run as $\text{time}(r) = \sum_{i=1}^n t_i$. We write Runs_{fin} for the set of finite runs.

3.2 Strategies

An average-time game Γ is a triple $(\mathcal{T}, L_{\text{Min}}, L_{\text{Max}})$, where $\mathcal{T} = (L, C, S, A, E, \delta, \varrho)$ is a timed automaton and $(L_{\text{Min}}, L_{\text{Max}})$ is a partition of L . We define $Q_{\text{Min}} = \{(\ell, \nu) \in Q : \ell \in L_{\text{Min}}\}$, $Q_{\text{Max}} = Q \setminus Q_{\text{Min}}$, $S_{\text{Min}} = S \cap Q_{\text{Min}}$, $S_{\text{Max}} = S \setminus S_{\text{Min}}$, $\mathcal{R}_{\text{Min}} = \{[s] : s \in Q_{\text{Min}}\}$, and $\mathcal{R}_{\text{Max}} = \mathcal{R} \setminus \mathcal{R}_{\text{Min}}$.

A *strategy* for Min is a function $\mu : \text{Runs}_{\text{fin}} \rightarrow A \times \mathbb{R}_\oplus$, such that if $\text{last}(r) = s \in S_{\text{Min}}$ and $\mu(r) = \tau$ then $s \xrightarrow{\tau} s'$, where $s' = \text{succ}(s, \tau)$. Similarly, a strategy for player Max is a function $\chi : \text{Runs}_{\text{fin}} \rightarrow A \times \mathbb{R}_\oplus$, such that if $\text{last}(r) = s \in S_{\text{Max}}$ and $\chi(r) = \tau$ then $s \xrightarrow{\tau} s'$, where $s' = \text{succ}(s, \tau)$. We write Σ_{Min} for the set of strategies for player Min, and we write Σ_{Max} for the set of strategies for player Max. If players Min and Max use strategies μ and χ , resp., then the (μ, χ) -run from a state s is the unique run $\text{run}(s, \mu, \chi) = \langle s_0, \tau_1, s_1, \tau_2, \dots \rangle$, such that $s_0 = s$, and for every $i \geq 1$, if $s_i \in S_{\text{Min}}$, or $s_i \in S_{\text{Max}}$, then $\mu(\text{run}_i(s, \mu, \chi)) = \tau_{i+1}$, or $\chi(\text{run}_i(s, \mu, \chi)) = \tau_{i+1}$, resp., where $\text{run}_i(s, \mu, \chi) = \langle s_0, \tau_1, s_1, \dots, s_{i-1}, \tau_i, s_i \rangle$.

We say that a strategy μ for Min is *positional* if for all finite runs $r, r' \in \text{Runs}_{\text{fin}}$, we have that $\text{last}(r) = \text{last}(r')$ implies $\mu(r) = \mu(r')$. A positional strategy for player Min can be then represented as a function $\mu : S_{\text{Min}} \rightarrow A \times \mathbb{R}_{\oplus}$, which uniquely determines the strategy $\mu^{\infty} \in \Sigma_{\text{Min}}$ as follows: $\mu^{\infty}(r) = \mu(\text{last}(r))$, for all finite runs $r \in \text{Runs}_{\text{fin}}$. Positional strategies for player Max are defined and represented in the analogous way. We write Π_{Min} and Π_{Max} for the sets of positional strategies for player Min and for player Max, respectively.

3.3 Value of Average-Time Game

If player Min uses the strategy $\mu \in \Sigma_{\text{Min}}$ and player Max uses the strategy $\chi \in \Sigma_{\text{Max}}$ then player Min loses the value

$$\mathcal{A}_{\text{Min}}(s, \mu, \chi) = \limsup_{n \rightarrow \infty} \frac{1}{n} \cdot \text{time}(\text{run}_n(s, \mu, \chi)),$$

and player Max wins the value

$$\mathcal{A}_{\text{Max}}(s, \mu, \chi) = \liminf_{n \rightarrow \infty} \frac{1}{n} \cdot \text{time}(\text{run}_n(s, \mu, \chi)).$$

In an average-time game player Min is interested in minimising the value she loses and player Max is interested in maximising the value he wins. For every state $s \in S$ of a timed automaton, we define its *upper value* by

$$\overline{\text{val}}^T(s) = \inf_{\mu \in \Sigma_{\text{Min}}} \sup_{\chi \in \Sigma_{\text{Max}}} \mathcal{A}_{\text{Min}}(s, \mu, \chi),$$

and its lower value

$$\underline{\text{val}}^T(s) = \sup_{\chi \in \Sigma_{\text{Max}}} \inf_{\mu \in \Sigma_{\text{Min}}} \mathcal{A}_{\text{Max}}(s, \mu, \chi).$$

The inequality $\underline{\text{val}}^T(s) \leq \overline{\text{val}}^T(s)$ always holds. An average-time game is *determined* if for every state $s \in S$, its lower and upper values are equal to each other; then we say that the *value* $\text{val}^T(s)$ exists and $\text{val}^T(s) = \underline{\text{val}}^T(s) = \overline{\text{val}}^T(s)$. We give an elementary proof for the determinacy of the average-time games without recourse to general results like Martin's determinacy theorem [14,15].

Theorem 2 (Determinacy). *Average-time games are determined.*

For strategy $\mu \in \Sigma_{\text{Min}}$ of player Min and $\chi \in \Sigma_{\text{Max}}$ of player Max, we define $\text{val}^{\mu}(s) = \sup_{\chi \in \Sigma_{\text{Max}}} \mathcal{A}_{\text{Min}}(s, \mu, \chi)$, and $\text{val}^{\chi}(s) = \inf_{\mu \in \Sigma_{\text{Min}}} \mathcal{A}_{\text{Max}}(s, \mu, \chi)$. For an $\varepsilon > 0$, we say that a strategy $\mu \in \Sigma_{\text{Min}}$ or $\chi \in \Sigma_{\text{Max}}$ is ε -*optimal* if for every $s \in S$ we have that $\text{val}^{\mu}(s) \leq \text{val}^T(s) + \varepsilon$ or $\text{val}^{\chi}(s) \geq \text{val}^T(s) - \varepsilon$, respectively. Note that if a game is determined then for every $\varepsilon > 0$, both players have ε -optimal strategies.

We say that a strategy $\chi \in \Sigma_{\text{Max}}$ of player Max is a best response to a strategy $\mu \in \Sigma_{\text{Min}}$ of player Min if for all $s \in S$ we have that $\mathcal{A}_{\text{Min}}(s, \mu, \chi) =$

$\sup_{\chi' \in \Sigma_{\text{Max}}} \mathcal{A}_{\text{Min}}(s, \mu, \chi')$. Similarly we say that a strategy $\mu \in \Sigma_{\text{Min}}$ of player Min is a best response to a strategy $\chi \in \Sigma_{\text{Max}}$ of player Max if for all $s \in S$ we have that $\mathcal{A}_{\text{Max}}(s, \mu, \chi) = \inf_{\mu' \in \Sigma_{\text{Min}}} \mathcal{A}_{\text{Max}}(s, \mu', \chi)$.

In the next section we introduce some region-based abstractions of timed automata, including the closed region graph, and its subgraphs: the boundary region graph, and the region graph. While the region graph is semantically equivalent to the corresponding timed automaton, the boundary region graph has the property that for every starting state, the reachable state space is finite. In Section 6 we introduce average-time games on these graphs and show that if we have the solution of the average-time game for any of these graphs, then we get the solution of the average-time game for the corresponding timed automaton. The key Theorem 2 follows immediately from Theorem 3.

4 Abstractions of Timed Automata

The region automaton, originally proposed by Alur and Dill [2], is a useful abstraction of a timed automaton as it preserves the validity of qualitative reachability, safety, and ω -regular properties. The *region automaton* [2] $\text{RA}(\mathcal{T}) = (\mathcal{R}, \mathcal{M})$ of a timed automaton \mathcal{T} consists of:

- the set \mathcal{R} of regions of \mathcal{T} , and
- $\mathcal{M} \subseteq \mathcal{R} \times (\mathcal{R} \times A) \times \mathcal{R}$, such that for all $a \in A$, and for all $R, R', R'' \in \mathcal{R}$, we have that $(R, R'', a, R') \in \mathcal{M}$ iff $R \rightarrow_* R'' \xrightarrow{a} R'$.

The region automaton, however, is not sufficient for solving average-time games as it abstracts away the timing information. Corner-point abstraction, introduced by Bouyer et al. [5], is a refinement of region automaton which preserves some timing information. Formally, the corner-point abstraction $\text{CP}(\mathcal{T})$ of a timed automaton \mathcal{T} is a finite graph (V, E) such that:

- $V \subseteq Q \times \mathcal{R}$ such that $(s, R) \in V$ iff $s = (\ell, \nu) \in \text{clos}(R)$ and ν is a corner. Since timed automata we consider are bounded, there are finitely many regions, and every region has a finite number of corners. Hence the set of vertices finite.
- $E \subseteq V \times (\mathbb{R}_{\oplus} \times \mathcal{R} \times A) \times V$ such that for $(s, R), (s', R') \in V$ and $(t, R'', a) \in \mathbb{R}_{\oplus} \times \mathcal{R} \times A$, we have $((s, R), (t, R'', a), (s', R')) \in E$ iff $R \rightarrow_* R'' \xrightarrow{a} R'$ and $(s + t) \xrightarrow{a} s'$. Notice that such a t is always a natural number.

Bouyer et al. [5] showed that the corner-point abstraction is sufficient for deciding one-player average-price problem if the initial state is a corner-state, i.e., a state whose clock valuation is a corner.

In this section we introduce the *boundary region graph*, which is a generalisation of the corner-point abstraction. We prove that the value of the average-time game on a timed automaton is equal to the value of the average-time game on the corresponding boundary region graph, for all starting states, not just for corner states. In the process, we introduce two other refinements of the region

automaton, which we call the *closed region graph* and the *region graph*. We collectively refer to these three graphs as *region graphs*. The analysis of average-time games on those objects allows us to establish equivalence of average-time games on the original timed automaton and the boundary region graph. We also show (Lemma 1) that the value of an average-time game is constant over a region. A side-effect of this result is that the corner-point abstractions can be used to solve average-time games on timed automata for arbitrary starting states.

4.1 Region Graphs

A *configuration* in region graphs is a pair (s, R) , where $s \in Q$ is a configuration of the timed automaton and $R \in \mathcal{R}$ is a region; We write Ω for the set of configurations of the region graphs. For a set $X \subseteq \Omega$ and a region $R_0 \in \mathcal{R}$, we define the set X restricted to the region R_0 as the set $\{(s, R) \in X : R = R_0\}$, and we denote this set by $X(R_0)$. For a configuration $q = (s, R) \in \Omega$ we write $[q]$ for its region R .

Definition 1 (Closed Region Graph). *The closed region graph $\overline{\mathcal{T}} = (\overline{S}, \overline{E})$ of a timed automaton \mathcal{T} is a labelled transition system, where:*

- \overline{S} is the set of states defined as

$$\overline{S} = \{(s, R) \in \Omega : s \in \text{clos}(R)\} \quad \text{and}$$

- \overline{E} is the labelled transition relation defined as

$$\begin{aligned} \overline{E} = \{((s, R), (t, R'', a), (s', R')) \in \overline{S} \times (\mathbb{R}_{\oplus} \times \mathcal{R} \times A) \times \overline{S} \\ : R \rightarrow_* R'' \xrightarrow{a} R' \text{ and } s' = \text{succ}(s, (t, a)) \text{ and } s + t \in \text{clos}(R'')\}. \end{aligned}$$

Definition 2 (Boundary Region Graph). *The boundary region graph $\widehat{\mathcal{T}} = (\widehat{S}, \widehat{E})$ of a timed automaton \mathcal{T} is a labelled transition system, where:*

- \widehat{S} is the set of states defined as

$$\widehat{S} = \{(s, R) \in \Omega : s \in \text{clos}(R)\} \quad \text{and}$$

- \widehat{E} is the labelled transition relation defined as

$$\begin{aligned} \widehat{E} = \{((s, R), (t, R'', a), (s', R')) \in \widehat{S} \times (\mathbb{R}_{\oplus} \times \mathcal{R} \times A) \times \widehat{S} \\ : R \rightarrow_* R'' \xrightarrow{a} R' \text{ and } s' = \text{succ}(s, (t, a)) \text{ and } s + t \in \text{bd}(R'')\}. \end{aligned}$$

Boundary region graphs have the following remarkable property.

Proposition 1 ([17]). *For every configuration in a boundary region graph the set of reachable configurations is finite.*

We say that a configuration $q = (s = (\ell, \nu), R)$ is *corner configuration* if ν is a corner.

Proposition 2. *The reachable sub-graph of the a boundary region graph \widehat{T} from a corner configuration is same as the corner-point abstraction $CP(T)$.*

Definition 3 (Region Graph). *A region graph of a timed automaton T is a labelled transition system $\widetilde{T} = (\widetilde{S}, \widetilde{E})$, where:*

- \widetilde{S} is the set of states defined as

$$\widetilde{S} = \{(s, R) \in \Omega : s \in R\} \text{ and}$$

- \widetilde{E} is the labelled transition relation defined as

$$\begin{aligned} \widetilde{E} = \{((s, R), (t, R'', a), (s', R')) \in \widetilde{S} \times (\mathbb{R}_{\oplus} \times \mathcal{R} \times A) \times \widetilde{S} \\ : R \rightarrow_* R'' \xrightarrow{a} R' \text{ and } s' = \text{succ}(s, (t, a)) \text{ and } s + t \in R''\}. \end{aligned}$$

For configuration $q = (s, R) \in \Omega$, real number $t \in \mathbb{R}_{\oplus}$, region $R'' \in \mathcal{R}$, and action $a \in A$, we write $\text{succ}(q, (t, R'', a))$ for the configuration $(\text{succ}(s, (t, a)), R')$ where $R'' \xrightarrow{a} R'$.

4.2 Region Game Graphs

For $\Gamma = (T, L_{\text{Min}}, L_{\text{Max}})$ we define the sets $\Omega_{\text{Min}} = \{(s, R) \in \Omega : R \in \mathcal{R}_{\text{Min}}\}$ and $\Omega_{\text{Max}} = \Omega \setminus \Omega_{\text{Min}}$. Similarly we define sets $\overline{S}_{\text{Min}}, \overline{S}_{\text{Max}}, \widehat{S}_{\text{Min}}, \widehat{S}_{\text{Max}}, \widetilde{S}_{\text{Min}},$ and $\widetilde{S}_{\text{Max}}$. The timed game automaton Γ naturally gives rise to the closed region game graph $\overline{T} = (\overline{T}, \overline{S}_{\text{Min}}, \overline{S}_{\text{Max}})$, the boundary region game graph $\widehat{T} = (\widehat{T}, \widehat{S}_{\text{Min}}, \widehat{S}_{\text{Max}})$, and the region game graph $\widetilde{T} = (\widetilde{T}, \widetilde{S}_{\text{Min}}, \widetilde{S}_{\text{Max}})$. When it is clear from context, we use the terms region graphs and region game graphs interchangeably. Also, sometimes, we write $T, \overline{T}, \widehat{T}$, and \widetilde{T} for $\Gamma, \overline{T}, \widehat{T}$, and \widetilde{T} , respectively.

4.3 Runs of Region Graphs

An infinite run of the closed region graph $\overline{T} = (\overline{S}, \overline{E})$ is an infinite sequence

$$\langle q_0, \tau_1, q_1, \tau_1, \dots \rangle \in \overline{S} \times ((\mathbb{R}_{\oplus} \times \mathcal{R} \times A) \times \overline{S})^{\omega},$$

such that for every positive integer i we have $(q_{i-1}, \tau_i, q_i) \in \overline{E}$. A finite run of the closed region graph \overline{T} is a finite sequence

$$\langle q_0, \tau_1, q_1, \tau_1, \dots, q_n \rangle \in \overline{S} \times ((\mathbb{R}_{\oplus} \times \mathcal{R} \times A) \times \overline{S})^*,$$

such that for every positive integer $i \leq n$ we have $(q_{i-1}, \tau_i, q_i) \in \overline{E}$. Runs of the boundary region graph and the region graph are defined analogously.

For a graph $\mathcal{G} \in \{\overline{T}, \widehat{T}, \widetilde{T}\}$ we write $\text{Runs}^{\mathcal{G}}$ for the set of its runs and $\text{Runs}^{\mathcal{G}}(q)$ for the set of its runs from a state $q \in \overline{Q}$. We write $\text{Runs}_{\text{fin}}^{\mathcal{G}}$ for the set of finite runs and $\text{Runs}_{\text{fin}}^{\mathcal{G}}(q)$ for the set of finite runs starting from $q \in \overline{S}$.

4.4 Pre-Runs and Run Types

Pre-runs [12] generalise runs of $\overline{\mathcal{T}}$, $\widetilde{\mathcal{T}}$, and $\widehat{\mathcal{T}}$, and allow us to compare the runs in $\overline{\mathcal{T}}$, $\widetilde{\mathcal{T}}$, and $\widehat{\mathcal{T}}$ in a uniform manner. On the other hand, the concept of the type [12] of a run allows us to compare pre-runs passing through the same sequence of regions.

A *pre-run* is a sequence $\langle (s_0, R_0), (t_1, R'_1, a_1), (s_1, R_1), \dots \rangle \in \Omega \times ((\mathbb{R}_{\oplus} \times \mathcal{R} \times A) \times \Omega)^\omega$, such that $s_{i+1} = \text{succ}(s_i, (t_{i+1}, a_{i+1}))$ and $R_i \rightarrow_* R'_{i+1} \xrightarrow{a_{i+1}} R_{i+1}$ for every $i \in \mathbb{N}$. We write PreRuns for the set of pre-runs and $\text{PreRuns}(s, R)$ for the set of pre-runs starting from $(s, R) \in \Omega$. The relation between various sets of runs is as follows: for all $q \in \overline{Q}$ we have

$$\begin{aligned} \text{Runs}^{\widehat{\mathcal{T}}}(q) &\subseteq \text{Runs}^{\overline{\mathcal{T}}}(q) \subseteq \text{PreRuns}(q) \quad \text{and} \\ \text{Runs}^{\widetilde{\mathcal{T}}}(q) &\subseteq \text{Runs}^{\overline{\mathcal{T}}}(q) \subseteq \text{PreRuns}(q). \end{aligned}$$

A finite pre-run is a finite sequence $\langle (s_0, R_0), (t_1, R'_1, a_1), \dots, (s_n, R_n) \rangle \in (Q \times \mathcal{R}) \times ((\mathbb{R}_{\oplus} \times \mathcal{R} \times A) \times (Q \times \mathcal{R}))^*$ such that for every nonnegative integer $i < n$ we have that $s_{i+1} = \text{succ}(s_i, (t_{i+1}, a_{i+1}))$ and $R_i \rightarrow_* R'_i \xrightarrow{a_{i+1}} R_{i+1}$. We write $\text{PreRuns}_{\text{fin}}$ for the set of finite pre-runs and $\text{PreRuns}_{\text{fin}}(s, R)$ for the set of finite pre-runs starting from $(s, R) \in \Omega$. For finite run $r = \langle q_0, (t_1, R_1, a_1), q_1, \dots, q_n \rangle \in \text{PreRuns}_{\text{fin}}$ we define its total time as $\text{time}(r) = \sum_{i=1}^n t_i$, and we denote the last state of the run by $\text{last}(r) = q_n$.

A *run type* is a sequence $\langle R_0, (R'_1, a_1), R_1, (R'_2, a_2), \dots \rangle \in \mathcal{R} \times ((\mathcal{R} \times A) \times \mathcal{R})^\omega$ such that for every $i \in \mathbb{N}$ we have that $R_i \rightarrow_* R'_{i+1} \xrightarrow{a_{i+1}} R_{i+1}$. We say that a pre-run $r = \langle (s_0, R_0), (t_1, R'_1, a_1), (s_1, R_1), (t_2, R'_2, a_2), \dots \rangle$ is of the type $\langle R_0, (R'_1, a_1), R_1, (R'_2, a_2), \dots \rangle$. We say that a run $r = \langle s_0, (t_1, a_1), s_1, (t_2, a_2), \dots \rangle$ of a timed automaton \mathcal{T} is of the type $\langle R_0, (R'_1, a_1), R_1, (R'_2, a_2), \dots \rangle$, where $R_i = [s_i]$ and $R'_{i+1} = [s_i + t_{i+1}]$ for all $i \in \mathbb{N}$. We also define the type of a finite runs analogously.

For a (finite or infinite) run or pre-run r , we write $\llbracket r \rrbracket_{\mathcal{R}}$ for its type. We write Types for the set of run types, and we write $\text{Types}(R)$ for the set of run types starting from region $R \in \mathcal{R}$. Similarly we write $\text{Types}_{\text{fin}}$ for the set of finite run types, and we write $\text{Types}_{\text{fin}}(R)$ for the set of finite run types starting from region $R \in \mathcal{R}$.

5 Strategies in Region Graphs

In this section we define strategies of players in region graphs $\overline{\mathcal{T}}$, $\widetilde{\mathcal{T}}$, and $\widehat{\mathcal{T}}$, and study some of their properties. Strategies in $\widetilde{\mathcal{T}}$ are called *admissible strategies*, while strategies in $\widehat{\mathcal{T}}$ are called *boundary strategies*. We also introduce so-called *type-preserving boundary strategies* which are a key tool in proving the correctness of game reduction from timed automata to boundary region graph. In Section 6 we show that there are optimal type-preserving boundary strategies in $\overline{\mathcal{T}}$ and $\widehat{\mathcal{T}}$.

5.1 Pre-strategies and Strategies in $\overline{\mathcal{T}}, \widehat{\mathcal{T}}, \widetilde{\mathcal{T}}$

Pre-strategies generalise the concept of strategies in region graphs, and allows us to discuss the strategies in $\overline{\mathcal{T}}, \widehat{\mathcal{T}}$, and $\widetilde{\mathcal{T}}$ in a uniform manner. We first define pre-strategies for players in \mathcal{T} , and then using that we define strategies for players in closed region graph, boundary region graph, and region graph.

Definition 4 (Pre-strategies). A pre-strategy of player Min μ is a (partial) function $\mu : \text{PreRuns}_{\text{fin}} \rightarrow \mathbb{R}_{\oplus} \times \mathcal{R} \times A$, such that for a run $r \in \text{PreRuns}_{\text{fin}}$, if $\text{last}(r) = (s, R) \in \Omega_{\text{Min}}$ then $\mu(r) = (t, R', a)$ is defined, and it is such that $R \rightarrow_* R'' \xrightarrow{a} R'$ for some $R' \in \mathcal{R}$. Pre-strategies of player Max are defined analogously. We write $\Sigma_{\text{Min}}^{\text{pre}}$ and $\Sigma_{\text{Max}}^{\text{pre}}$ for the set of pre-strategies of player Min and player Max, respectively.

We say that a strategy of player Min $\mu \in \Sigma_{\text{Min}}^{\text{pre}}$ is positional if for all runs $r_1, r_2 \in \text{PreRuns}_{\text{fin}}$ we have that $\text{last}(r_1) = \text{last}(r_2)$ implies $\mu(r_1) = \mu(r_2)$. Similarly we define positional strategy of player Max.

We define the run starting from configuration $q \in \Omega$ where player Min and player Max use the strategies $\mu \in \Sigma_{\text{Min}}^{\text{pre}}$ and $\chi \in \Sigma_{\text{Max}}^{\text{pre}}$, respectively, in a straightforward manner and we write $\text{run}(q, \mu, \chi)$ for this run. For every positive integer n we write $\text{run}_n(q, \mu, \chi)$ for the prefix of the run $\text{run}(q, \mu, \chi)$ of length n .

Now we are in a position to introduce strategies in closed region graph, region graph, and boundary region graph.

Definition 5 (Strategies in Closed Region Graph). A pre-strategy of player Min $\mu \in \Sigma_{\text{Min}}^{\text{pre}}$ is a strategy in a closed region graph $\overline{\mathcal{T}} = (\overline{S}, \overline{E})$ if for every run $r \in \text{PreRuns}_{\text{fin}}$ such that $\mu(r) = (t, R', a)$, we have that $(s + t) \in \text{clos}(R')$ where $(s, R) = \text{last}(r)$. Strategies of player Max in a closed region graph are defined analogously. We write $\overline{\Sigma}_{\text{Min}}$ and $\overline{\Sigma}_{\text{Max}}$ for the set of strategies of player Min and player Max, respectively.

Definition 6 (Strategies in Region Graphs). A pre-strategy of player Min $\mu \in \Sigma_{\text{Min}}^{\text{pre}}$ is a strategy in a region graph $\widetilde{\mathcal{T}} = (\widetilde{S}, \widetilde{E})$ if for every run $r \in \text{Runs}_{\text{fin}}^{\widetilde{\mathcal{T}}}$ such that $\mu(r) = (t, R', a)$, we have that $(s + t) \in R'$ where $(s, R) = \text{last}(r)$. Strategies of player Max in a region graph are defined analogously. We call such strategies admissible strategies. We write $\widetilde{\Sigma}_{\text{Min}}$ and $\widetilde{\Sigma}_{\text{Max}}$ for the set of admissible strategies of player Min and player Max, respectively.

Definition 7 (Strategies in Boundary Region Graph). A pre-strategy of player Min $\mu \in \Sigma_{\text{Min}}^{\text{pre}}$ is a strategy in a boundary region graph $\widehat{\mathcal{T}} = (\widehat{S}, \widehat{E})$ if for every run $r \in \text{PreRuns}_{\text{fin}}$ such that $\mu(r) = (t, R', a)$, we have that

$$t = \inf\{t : s + t \in \text{clos}(R')\}, \quad (1)$$

where $(s, R) = \text{last}(r)$.

A pre-strategy of player Max $\chi \in \Sigma_{\text{Max}}^{\text{pre}}$ is a strategy in a boundary region graph $\widehat{\mathcal{T}}$ if for every run $r \in \text{PreRuns}_{\text{fin}}$ such that $\mu(r) = (t, R', a)$, we have that

$$t = \sup\{t : s + t \in \text{clos}(R')\}, \quad (2)$$

where $(s, R) = \text{last}(r)$. We call such strategies boundary strategies. We write $\widehat{\Sigma}_{\text{Min}}$ and $\widehat{\Sigma}_{\text{Max}}$ for the set of boundary strategies of player Min and player Max, respectively.

For notational convenience and w.l.o.g., in the definition of boundary strategies, we do not consider those timed moves of player Min (Max) which suggest waiting till the farther (nearer) boundary of a thick region.

Remark 1. For every state $s \in S$ of timed automata \mathcal{T} and every strategy $\mu \in \Sigma_{\text{Min}}^{\text{pre}}$ and $\chi \in \Sigma_{\text{Max}}^{\text{pre}}$ of respective players, we have that :

- $\text{run}((s, [s]), \mu, \chi) \in \text{Runs}^{\overline{\mathcal{T}}}(s, [s])$ if $\mu \in \overline{\Sigma}_{\text{Min}}$ and $\chi \in \overline{\Sigma}_{\text{Max}}$;
- $\text{run}((s, [s]), \mu, \chi) \in \text{Runs}^{\widehat{\mathcal{T}}}(s, [s])$ if $\mu \in \widehat{\Sigma}_{\text{Min}}$ and $\chi \in \widehat{\Sigma}_{\text{Max}}$;
- $\text{run}((s, [s]), \mu, \chi) \in \text{Runs}^{\widetilde{\mathcal{T}}}(s, [s])$ if $\mu \in \widetilde{\Sigma}_{\text{Min}}$ and $\chi \in \widetilde{\Sigma}_{\text{Max}}$.

Boundary Strategies and Boundary Timed Actions. Define the finite set of *boundary timed actions* $\mathbb{A} = ([k]_{\mathbb{N}} \times C \times A$. For $s \in Q$ and $\alpha = (b, c, a) \in \mathbb{A}$, we define $t(s, \alpha) = b - s(c)$ if $s(c) \leq b$, and $t(s, \alpha) = 0$ if $s(c) > b$; and we define $\text{succ}(s, \alpha)$ to be the state $s' = \text{succ}(s, \tau(\alpha))$, where $\tau(\alpha) = (t(s, \alpha), a)$; we then write $s \xrightarrow{\alpha} s'$. We also write $s \xrightarrow{\alpha} s'$ if $s \xrightarrow{\tau(\alpha)} s'$. For configuration $q = (s, R) \in \Omega$, boundary timed action $\alpha = (b, c, a) \in \mathbb{A}$, and region $R'' \in \mathcal{R}$ we write $\text{succ}(q, (\alpha, R''))$ for the configuration $\text{succ}(q, (t(s, \alpha), R'', a))$.

Timed actions suggested by a boundary strategies are precisely boundary timed actions. The following proposition formalises this notion.

Proposition 3. *For every boundary strategy $\sigma \in \widehat{\Sigma}_{\text{Min}}(\widehat{\Sigma}_{\text{Max}})$ of player Min (Max) and for every run $r \in \text{PreRuns}_{\text{fin}}$, if $\sigma(r) = (t, R', a)$ then there exists a boundary timed action $\alpha = (b, c, a) \in \mathbb{A}$ such that $t(s, \alpha) = t$, where $(s, R) = \text{last}(r)$.*

Proof. Let run $r \in \text{PreRuns}_{\text{fin}}$ be such that $\text{last}(r) = (s, R)$. Let $\sigma \in \widehat{\Sigma}_{\text{Min}}$ be a boundary strategy of player Min such that $\sigma(r) = (t, R', a)$. From the definition of the boundary strategies, we have that $t = \inf\{t : s + t \in \text{clos}(R')\}$. To prove the proposition, all we need to show is that there exists an integer $b \in \mathbb{Z}$ and a clock $c \in C$, such that $b - s(c) = t$.

If $R' \in \mathcal{R}_{\text{Thin}}$ then there exists a clock $c' \in C$ such that for all states $s' \in \text{clos}(R')$ we have that $\lceil s'(c') \rceil = 0$. In this case the clock $c = c'$ and the integer $b = (s + t)(c)$.

If $R \in \mathcal{R}_{\text{Thick}}$ and let $R' \leftarrow_{+1} R$ be the thin region immediately before R . Let clock $c' \in C$ be such that for all states $s' \in \text{clos}(R')$ we have that $\lceil s'(c') \rceil = 0$. Again, in this case the desired clock $c = c'$ and the integer $b = (s + t)(c)$.

The case, where σ is a strategy of Max is similar, and hence omitted. \square

Sometimes, in our proofs we need to use boundary timed action suggested by a boundary strategy. For this purpose we define the notation $\widehat{\sigma}(r)$ that gives the boundary timed action and region pair that corresponds to $\sigma(r)$. The definition of this function is formalised in the following definition.

Definition 8. For a boundary strategy $\sigma \in \hat{\Sigma}_{Max}(\hat{\Sigma}_{Max})$ of player Min (Max), we define the function $\hat{\sigma} : PreRuns_{fin} \rightarrow (\mathbb{A} \times \mathcal{R})$ as follows: if for a run $r \in PreRuns_{fin}$ we have $\sigma(r) = (t, R', a)$, then $\hat{\sigma}(r) = ((b, c, a), R')$ such that $b-s(c) = t$, where $(s, R) = last(r)$.

5.2 Type-Preserving Boundary Strategies

We now introduce an important class of boundary strategies called *type-preserving boundary strategies*. Broadly speaking, these strategies suggest to players a unique boundary timed action and region pair for all the finite runs of the same type.

Definition 9 (Type-Preserving Boundary Strategies). A boundary strategy $\sigma \in \hat{\Sigma}_{Min}$ of player Min is *type-preserving* if $\llbracket r_1 \rrbracket_{\mathcal{R}} = \llbracket r_2 \rrbracket_{\mathcal{R}}$ implies $\hat{\sigma}(r_1) = \hat{\sigma}(r_2)$ for all $r_1, r_2 \in PreRuns_{fin}$. Type-preserving boundary strategies of player Max are defined analogously. We write Ξ_{Min} and Ξ_{Max} for the sets of type-preserving boundary strategies of players Min and Max, respectively.

The rationale behind the name *type-preserving* is that if $\mu \in \Xi_{Min}$ and $\chi \in \Xi_{Max}$, then for every $R \in \mathcal{R}$ and for $q, q' \in \Omega(R)$, the run types of the resulting runs from q and q' are the same, i.e., $\llbracket run(q, \mu, \chi) \rrbracket_{\mathcal{R}} = \llbracket run(q', \mu, \chi) \rrbracket_{\mathcal{R}}$.

Simple Functions. Let $X \subseteq \Omega$. A function $F : \overline{Q} \rightarrow \mathbb{R}$ is *simple* [3,11] if either: there is $e \in \mathbb{Z}$, such that for every $q = (s, R) \in X$, we have $F(q) = e$; or there are $e \in \mathbb{Z}$ and $c \in C$, such that for every $q = (s, R) \in X$, we have $F(q) = e - s(c)$. We say that a function $F : X \rightarrow \mathbb{R}$ is *regionally simple* or *regionally constant*, respectively, if for every region $R \in \mathcal{R}$, the function F , over domain $X(R)$, is simple or constant, respectively.

For regions $R, R', R'' \in \mathcal{R}$ and boundary timed action $\alpha = (b, c, a) \in \mathbb{A}$, we write $R \xrightarrow{R''}_{\alpha} R'$ if one of the following holds:

- $R \rightarrow_{b,c} R'' \xrightarrow{a} R'$, or
- there is region $R''' \in \mathcal{R}_{Thin}$ such that $R \rightarrow_{b,c} R''' \rightarrow_{+1} R'' \xrightarrow{a} R'$, or
- there is a region $R''' \in \mathcal{R}_{Thin}$ such that $R \rightarrow_{b,c} R''' \leftarrow_{+1} R'' \xrightarrow{a} R'$.

Properties of Type-preserving Boundary Strategies. The next two proposition state that if both players play with type-preserving boundary strategies then for every $n \in \mathbb{N}$ the total time spent in n transitions is regionally simple (Proposition 4), and the average time of the infinite run is regionally constant (Proposition 5).

Proposition 4 (Type-preserving strategy pairs yield regionally simple time for finite runs). If $\mu \in \Xi_{Min}$, $\chi \in \Xi_{Max}$, and $n \in \mathbb{N}$, then $time(run_n(\cdot, \mu, \chi)) : \overline{Q} \rightarrow \mathbb{R}_{\oplus}$ is regionally simple.

Proposition 5 (Type-preserving strategy pairs yield regionally constant average time). If $\mu \in \Xi_{Min}$ and $\chi \in \Xi_{Max}$ then $\mathcal{A}_{Min}(\cdot, \mu, \chi) : \overline{Q} \rightarrow \mathbb{R}_{\oplus}$ and $\mathcal{A}_{Max}(\cdot, \mu, \chi) : \overline{Q} \rightarrow \mathbb{R}_{\oplus}$ are regionally constant.

Type-preserving Boundary Strategy that Agrees with a Boundary Strategy. Given an arbitrary boundary strategy σ and a configuration $q \in \overline{Q}$, sometimes we are interested in a type-preserving boundary strategy that agrees with σ for all the runs starting from q . We denote such a strategy by $\sigma^{\perp q}$. The following definition formalises such strategy.

Definition 10. For a boundary strategy $\mu \in \widehat{\Sigma}_{Min}$ of player Min and $q \in \overline{Q}$ we define $\mu^{\perp q} \in \Xi_{Min}$ to be a type-preserving boundary strategy which satisfy the following conditions:

1. $\widehat{\mu^{\perp q}}(r) = \widehat{\mu}(r)$ for every $r \in \text{PreRuns}_{fin}(q)$, and
2. $\llbracket r \rrbracket_{\mathcal{R}} = \llbracket r' \rrbracket_{\mathcal{R}}$ implies $\widehat{\mu^{\perp q}}(r) = \widehat{\mu^{\perp q}}(r')$ for all runs $r, r' \in \text{PreRuns}_{fin}$.

For $\chi \in \widehat{\Sigma}_{Max}$ and $q \in \overline{Q}$ we define $\chi^{\perp q} \in \Xi_{Max}$ analogously.

Given an arbitrary strategy $\mu \in \overline{\Sigma}_{Min}$ of player Min, a type-preserving boundary strategy $\chi \in \Xi_{Max}$ of player Max, and a configuration $q \in \overline{Q}$ sometimes we require to specify a type-preserving strategy $\mu^{(q, \chi)} \in \Xi_{Min}$ which has the property that types of runs $\text{run}(q, \mu, \chi)$ and $\text{run}(q, \mu^{(q, \chi)}, \chi)$ are the same. We then argue that from configuration $q \in \overline{Q}$ if player Max plays according to $\chi \in \Xi_{Max}$ then player Min can achieve better average-time if she plays according to $\mu^{(q, \chi)}$ (see Proposition 6 and Corollary 1). The motivation for the definition of $\chi^{(q, \mu)}$ is similar.

Definition 11. For an arbitrary strategy $\mu \in \overline{\Sigma}_{Min}$ of player Min, a type-preserving boundary strategy $\chi \in \Xi_{Max}$ of player Max, and a configuration $q = (s, R) \in \overline{Q}$, we define $\mu^{(q, \chi)} \in \Xi_{Min}$ to be a type-preserving boundary strategy which satisfy the following conditions:

1. $\llbracket \text{run}(q, \mu^{(q, \chi)}, \chi) \rrbracket_{\mathcal{R}} = \llbracket \text{run}(q, \mu, \chi) \rrbracket_{\mathcal{R}}$, and
2. $\llbracket r \rrbracket_{\mathcal{R}} = \llbracket r' \rrbracket_{\mathcal{R}}$ implies $\widehat{\mu^{(q, \chi)}}(r) = \widehat{\mu^{(q, \chi)}}(r')$ for all runs $r, r' \in \text{PreRuns}_{fin}$.

For $\chi \in \overline{\Sigma}_{Max}$, $\mu \in \Xi_{Min}$, and $q \in \overline{Q}$ the strategy $\chi^{(q, \mu)} \in \Xi_{Max}$ is defined analogously.

The following proposition and its corollary shows that starting from a configuration q player Min (Max) prefers $\mu^{(q, \chi)}$ ($\chi^{(q, \mu)}$) to μ (χ) against a type-preserving strategy $\chi \in \Xi_{Max}$ ($\mu \in \Xi_{Min}$) of its opponent.

Proposition 6. For every $\chi \in \Xi_{Max}$, $\mu \in \overline{\Sigma}_{Min}$ and $q \in \overline{Q}$ we have that

$$\text{time}(\text{run}_n(q, \mu, \chi)) \geq \text{time}(\text{run}_n(q, \mu^{(q, \chi)}, \chi)),$$

for every $n \in \mathbb{N}$. Similarly, for every $\mu \in \Xi_{Min}$, $\chi \in \overline{\Sigma}_{Max}$ and $q \in \overline{Q}$ we have that

$$\text{time}(\text{run}_n(q, \mu, \chi)) \leq \text{time}(\text{run}_n(q, \mu, \chi^{(q, \mu)})),$$

for every $n \in \mathbb{N}$.

An easy corollary of this proposition is as follows:

Corollary 1. For every $\chi \in \Xi_{Max}$, $\mu \in \overline{\Sigma}_{Min}$ and for all configurations $q \in \overline{Q}$ we have that

$$\mathcal{A}_{Min}(q, \mu, \chi) \geq \mathcal{A}_{Min}(q, \mu^{(q, \chi)}, \chi).$$

Similarly for every $\mu \in \Xi_{Min}$, $\chi \in \overline{\Sigma}_{Max}$ and for all configurations $q \in \overline{Q}$ we have that

$$\mathcal{A}_{Max}(q, \mu, \chi) \leq \mathcal{A}_{Max}(q, \mu, \chi^{(q, \mu)}).$$

Admissible Strategies ε -Close to a Type-Preserving Boundary Strategy. Given a type-preserving boundary strategy σ and a positive real $\varepsilon > 0$, sometimes we are interested in admissible strategies that behave like σ within ε precision. The following definition formalises such strategy.

Definition 12. For $\mu \in \Xi_{Min}$ and a real number $\varepsilon > 0$, we define the set of admissible strategy $\tilde{\Sigma}_{Min}^{(\mu, \varepsilon)} \subseteq \tilde{\Sigma}_{Min}$ as follows. For every $\mu_\varepsilon \in \tilde{\Sigma}_{Min}^{(\mu, \varepsilon)}$ we have that for all runs $r \in \text{PreRuns}_{fin}$ if $\hat{\mu}(r) = ((b, c, a), R')$ then $\mu_\varepsilon(r) = (t, R', a)$ is such that

$$s + t \in R' \text{ and } t \leq b - s(c) + \varepsilon,$$

where $(s, R) = \text{last}(r)$. Notice that (see Equation 1) such a value of t always exists.

Similarly for $\chi \in \Xi_{Max}$ and a real number $\varepsilon > 0$ we define the set $\tilde{\Sigma}_{Max}^{(\chi, \varepsilon)} \subseteq \tilde{\Sigma}_{Max}$ as follows. For every $\chi_\varepsilon \in \tilde{\Sigma}_{Max}^{(\chi, \varepsilon)}$ we have that for all runs $r \in \text{PreRuns}_{fin}$ if $\hat{\chi}(r) = ((b, c, a), R')$ then $\chi_\varepsilon(r) = (t, R', a)$ is such that

$$s + t \in R' \text{ and } t \geq b - s(c) - \varepsilon,$$

where $(s, R) = \text{last}(r)$.

Given an arbitrary strategy $\mu \in \overline{\Sigma}_{Min}$ of player Min, a positive real $\varepsilon > 0$, a type-preserving boundary strategy $\chi \in \Xi_{Max}$ of player Max, an ε -close strategy $\chi_\varepsilon \in \tilde{\Sigma}_{Max}^{(\chi, \varepsilon)}$, and a configuration $q \in \overline{Q}$ sometimes we require to specify a type-preserving strategy $\mu^{(q, \chi_\varepsilon)} \in \Xi_{Min}$ which has the property that types of runs $\text{run}(q, \mu, \chi_\varepsilon)$ and $\text{run}(q, \mu^{(q, \chi_\varepsilon)}, \chi_\varepsilon)$ are the same.

Definition 13. For an arbitrary strategy $\mu \in \overline{\Sigma}_{Min}$ of player Min, a positive real $\varepsilon > 0$, a type-preserving boundary strategy $\chi \in \Xi_{Max}$ of player Max, an ε -close strategy $\chi_\varepsilon \in \tilde{\Sigma}_{Max}^{(\chi, \varepsilon)}$, and a configuration $q = (s, R) \in \overline{Q}$, we define $\mu^{(q, \chi_\varepsilon)} \in \Xi_{Min}$ to be a type-preserving boundary strategy which satisfy the following conditions:

1. $\llbracket \text{run}(q, \mu^{(q, \chi_\varepsilon)}, \chi_\varepsilon) \rrbracket_{\mathcal{R}} = \llbracket \text{run}(q, \mu, \chi_\varepsilon) \rrbracket_{\mathcal{R}}$, and
2. $\llbracket r \rrbracket_{\mathcal{R}} = \llbracket r' \rrbracket_{\mathcal{R}}$ implies $\widehat{\mu^{(q, \chi_\varepsilon)}}(r) = \widehat{\mu^{(q, \chi_\varepsilon)}}(r')$ for all runs $r, r' \in \text{PreRuns}_{fin}$.

Combining it with Definition 12 we get that $\llbracket \text{run}(q, \mu^{(q, \chi_\varepsilon)}, \chi) \rrbracket_{\mathcal{R}} = \llbracket \text{run}(q, \mu, \chi_\varepsilon) \rrbracket_{\mathcal{R}}$. For $\chi \in \overline{\Sigma}_{Max}$, $\chi_\varepsilon \in \tilde{\Sigma}_{Max}^{(\chi, \varepsilon)}$, $\mu \in \Xi_{Min}$, and $q \in \overline{Q}$ the strategy $\chi^{(q, \mu_\varepsilon)} \in \Xi_{Max}$ is defined analogously.

We need the following property of $\mu^{(q, \chi_\varepsilon)}$ and $\chi^{(q, \mu_\varepsilon)}$ strategies.

Proposition 7. For every arbitrary strategy $\mu \in \overline{\Sigma}_{Min}$, positive real $\varepsilon > 0$, type-preserving boundary strategy $\chi \in \Xi_{Max}$ of player Max, ε -close strategy $\chi_\varepsilon \in \tilde{\Sigma}_{Max}^{(\chi, \varepsilon)}$ of player Max, and $q \in \overline{Q}$ we have

$$\text{time}(\text{run}_n(q, \mu, \chi_\varepsilon)) \geq \text{time}(\text{run}_n(q, \mu^{(q, \chi_\varepsilon)}, \chi)) - n \cdot \varepsilon,$$

for every $n \in \mathbb{N}$. Similarly for every arbitrary strategy $\chi \in \overline{\Sigma}_{Max}$, positive real $\varepsilon > 0$, type-preserving boundary strategy $\mu \in \Xi_{Min}$ of player Max, ε -close strategy $\mu_\varepsilon \in \tilde{\Sigma}_{Min}^{(\mu, \varepsilon)}$ of player Min, and $q \in \overline{Q}$ we have

$$\text{time}(\text{run}_n(q, \mu_\varepsilon, \chi)) \leq \text{time}(\text{run}_n(q, \mu_\varepsilon, \chi^{(q, \mu_\varepsilon)})) + n \cdot \varepsilon,$$

for every $n \in \mathbb{N}$.

The following result is an easy corollary of Proposition 7.

Corollary 2. For every $\chi \in \Xi_{Max}$, $\mu \in \overline{\Sigma}_{Min}$, $\varepsilon > 0$, $\chi_\varepsilon \in \tilde{\Sigma}_{Max}^{(\chi, \varepsilon)}$, and $q \in \overline{Q}$ we have that

$$\mathcal{A}_{Max}(q, \mu, \chi_\varepsilon) \geq \mathcal{A}_{Max}(q, \mu^{(q, \chi_\varepsilon)}, \chi) - \varepsilon.$$

Similarly for every $\mu \in \Xi_{Min}$, $\chi \in \overline{\Sigma}_{Max}$, $\varepsilon > 0$, $\mu_\varepsilon \in \tilde{\Sigma}_{Min}^{(\mu, \varepsilon)}$, and $q \in \overline{Q}$ we have that

$$\mathcal{A}_{Min}(q, \mu_\varepsilon, \chi) \leq \mathcal{A}_{Min}(q, \mu, \chi^{(q, \mu_\varepsilon)}) + \varepsilon.$$

To summarise the relations between various strategies, note that the following inclusions hold:

$$\begin{aligned} \Xi_{Min} \subseteq \hat{\Sigma}_{Min} \subseteq \overline{\Sigma}_{Min} \subseteq \Sigma_{Min}^{\text{pre}} \quad \text{and} \quad \tilde{\Sigma}_{Min} \subseteq \overline{\Sigma}_{Min} \subseteq \Sigma_{Min}^{\text{pre}}, \quad \text{and} \\ \Xi_{Max} \subseteq \hat{\Sigma}_{Max} \subseteq \overline{\Sigma}_{Max} \subseteq \Sigma_{Max}^{\text{pre}} \quad \text{and} \quad \tilde{\Sigma}_{Max} \subseteq \overline{\Sigma}_{Max} \subseteq \Sigma_{Max}^{\text{pre}}. \end{aligned}$$

6 Average-Time Games on Region Graphs

We define $\mathcal{A}_{Min} : \Omega \times \Sigma_{Min}^{\text{pre}} \times \Sigma_{Max}^{\text{pre}} \rightarrow \mathbb{R}_\oplus$ and $\mathcal{A}_{Max} : \Omega \times \Sigma_{Min}^{\text{pre}} \times \Sigma_{Max}^{\text{pre}} \rightarrow \mathbb{R}_\oplus$ in the following manner:

$$\begin{aligned} \mathcal{A}_{Min}(q, \mu, \chi) &= \limsup_{n \rightarrow \infty} \frac{1}{n} \cdot \text{time}(\text{run}_n(q, \mu, \chi)), \quad \text{and} \\ \mathcal{A}_{Max}(q, \mu, \chi) &= \liminf_{n \rightarrow \infty} \frac{1}{n} \cdot \text{time}(\text{run}_n(q, \mu, \chi)), \end{aligned}$$

where $\mu \in \Sigma_{Min}^{\text{pre}}$, $\chi \in \Sigma_{Max}^{\text{pre}}$ and $q \in \Omega$. For average-time games on a graph $\mathcal{G} \in \{\overline{\mathcal{T}}, \hat{\mathcal{T}}, \tilde{\mathcal{T}}\}$ we define the lower-value $\underline{\text{val}}^{\mathcal{G}}(q)$, the upper-value $\overline{\text{val}}^{\mathcal{G}}(q)$ and the value $\text{val}^{\mathcal{G}}(q)$ of a configuration $q \in \overline{Q}$ in a straightforward manner.

From construction it clear that the difference between an average-time game on a timed automaton and the average-time game on corresponding region graph is purely syntactical. Hence if the average-time game on region graph $\tilde{\mathcal{T}}$ is determined then average-time game on timed automaton \mathcal{T} is determined as well.

Proposition 8. *An average-time game on timed automaton \mathcal{T} is determined, if the corresponding average-time game on region graph $\tilde{\mathcal{T}}$ is determined. Moreover for all $s \in S$ we have that $\text{val}(s) = \text{val}^{\tilde{\mathcal{T}}}(s, [s])$.*

The following is the main result of this section.

Theorem 3. *Let \mathcal{T} be a timed automaton. Average-time games on the timed automaton \mathcal{T} , the closed region graph $\overline{\mathcal{T}}$, the region graph $\tilde{\mathcal{T}}$, and the boundary region graph $\hat{\mathcal{T}}$ are determined. Moreover for every $s \in S$ in a timed automaton \mathcal{T} , we have:*

$$\text{val}^{\mathcal{T}}(s) = \text{val}^{\tilde{\mathcal{T}}}(s, [s]) = \text{val}^{\overline{\mathcal{T}}}(s, [s]) = \text{val}^{\hat{\mathcal{T}}}(s, [s]).$$

This theorem follows from Theorem 4, Theorem 6, Theorem 7, and Proposition 8.

Moreover Theorem 3 and Proposition 5 let us conclude the following lemma about the value of average-time games on timed automata.

Lemma 1. *The value of every average-time game is regionally constant.*

An interesting implication of Lemma 1 is that corner-point abstraction is sufficient to solve average-time games with an arbitrary initial state.

6.1 Determinacy of Average-Time Games on the Boundary Region Graph

Positional determinacy of average-time games on the boundary region graph is immediate from Proposition 1 and Theorem 1.

Theorem 4. *The average-time game on $\hat{\mathcal{T}}$ is determined, and there are optimal positional strategies in $\hat{\mathcal{T}}$, i.e., for every $q \in \overline{Q}$, we have:*

$$\text{val}^{\hat{\mathcal{T}}}(q) = \inf_{\mu \in \hat{\Pi}_{Min}} \sup_{\chi \in \hat{\Sigma}_{Max}} \mathcal{A}_{Min}(q, \mu, \chi) = \sup_{\chi \in \hat{\Pi}_{Max}} \inf_{\mu \in \hat{\Sigma}_{Min}} \mathcal{A}_{Max}(q, \mu, \chi).$$

In fact, in a boundary region graph, there are optimal type-preserving boundary strategies. Before we show that, we need the following result.

Lemma 2. *In $\hat{\mathcal{T}}$, if $\mu \in \hat{\Sigma}_{Min}$ and $\chi \in \hat{\Sigma}_{Max}$ are mutual best responses from $q \in \overline{Q}$, then $\mu^{\downarrow q} \in \Xi_{Min}$ and $\chi^{\downarrow q} \in \Xi_{Max}$ are mutual best responses from every $q' \in \overline{Q}([q])$.*

Proof. We argue that $\chi^{\downarrow q}$ is a best response to $\mu^{\downarrow q}$ from $q' \in \overline{Q}([q])$ in $\hat{\mathcal{T}}$; the other case is analogous. For all $X \in \hat{\Sigma}_{Max}$, we have the following:

$$\begin{aligned} \mathcal{A}_{Min}(q', \mu^{\downarrow q}, \chi^{\downarrow q}) &= \mathcal{A}_{Min}(q, \mu^{\downarrow q}, \chi^{\downarrow q}) \geq \mathcal{A}_{Min}(q, \mu^{\downarrow q}, X^{\downarrow q'}) = \\ &= \mathcal{A}_{Min}(q', \mu^{\downarrow q}, X^{\downarrow q'}) = \mathcal{A}_{Min}(q', \mu^{\downarrow q}, X). \end{aligned}$$

The first equality follows from Proposition 5; the inequality follows because χ is a best response to μ from q ; the second equality follows from Proposition 5 again; and the last equality is straightforward. \square

Theorem 5. *There are optimal type-preserving boundary strategies in \widehat{T} , i.e., for every $q \in \overline{Q}$, we have:*

$$val^{\widehat{T}}(q) = \inf_{\mu \in \Xi_{Min}} \sup_{\chi \in \widehat{\Sigma}_{Max}} \mathcal{A}_{Min}(q, \mu, \chi) = \sup_{\chi \in \Xi_{Max}} \inf_{\mu \in \widehat{\Sigma}_{Min}} \mathcal{A}_{Max}(q, \mu, \chi).$$

Proof. Let $\mu^* \in \Xi_{Min}$ and $\chi^* \in \Xi_{Max}$ be mutual best responses in \widehat{T} ; existence of such strategies follows from Lemma 2. Moreover, we can assume that the strategies μ^* and χ^* have finite memory; this can be achieved by taking positional strategies $\mu \in \widehat{\Sigma}_{Min}$ and $\chi \in \widehat{\Sigma}_{Max}$ in Lemma 2. We then have the following:

$$\begin{aligned} \inf_{\mu \in \Xi_{Min}} \sup_{\chi \in \widehat{\Sigma}_{Max}} \mathcal{A}_{Min}(q, \mu, \chi) &\leq \sup_{\chi \in \widehat{\Sigma}_{Max}} \mathcal{A}_{Min}(q, \mu^*, \chi) = \mathcal{A}_{Min}(q, \mu^*, \chi^*) = \\ \mathcal{A}_{Max}(q, \mu^*, \chi^*) &= \inf_{\mu \in \widehat{\Sigma}_{Min}} \mathcal{A}_{Max}(q, \mu, \chi^*) \leq \sup_{\chi \in \Xi_{Max}} \inf_{\mu \in \widehat{\Sigma}_{Min}} \mathcal{A}_{Max}(q, \mu, \chi). \end{aligned}$$

The first and last inequalities are straightforward as $\mu^* \in \Xi_{Min}$ and $\chi^* \in \Xi_{Max}$. The first equality holds because χ^* is a best response to μ^* in \widehat{T} , and the third equality holds because μ^* is a best response to χ^* in \widehat{T} . Finally, the second equality holds because strategies μ^* and χ^* have finite memory. \square

6.2 Determinacy of Average-Time Games on the Closed Region Graph

To be able to show the determinacy of the average-time games on the closed region graph, we need the following intermediate result.

Lemma 3. *In \overline{T} , for every strategy in Ξ_{Min} there is a best response in Ξ_{Max} , and for every strategy in Ξ_{Max} there is a best response in Ξ_{Min} .*

Proof. We argue that if $\mu \in \overline{\Sigma}_{Min}$ is best-response to $\chi \in \Xi_{Max}$ from $q \in \overline{Q}$ then the strategy $\mu^{(q, \chi)}$ is best-response to χ from every $q' \in \overline{Q}([q])$. For all $M \in \overline{\Sigma}_{Min}$ we have the following:

$$\begin{aligned} \mathcal{A}_{Min}(q', \mu^{(q, \chi)}, \chi) &= \mathcal{A}_{Min}(q, \mu^{(q, \chi)}, \chi) \leq \mathcal{A}_{Min}(q, \mu, \chi) \leq \mathcal{A}_{Min}(q, M^{(q', \chi)}, \chi) = \\ &\mathcal{A}_{Min}(q', M^{(q', \chi)}, \chi) \leq \mathcal{A}_{Min}(q', \mu', \chi). \end{aligned}$$

The first and the second equalities follow from Proposition 5; the second inequality follows because μ is a best response to χ from q ; and the first and the third inequalities follow from the Corollary 1. It follows that in \overline{T} for every strategy $\chi \in \Xi_{Max}$ there is a best response in Ξ_{Min} . Similarly we prove that in \overline{T} for every strategy $\mu \in \Xi_{Min}$ there is a best response in Ξ_{Max} . \square

Theorem 6. *The average-time game on \overline{T} is determined, and there are optimal type-preserving boundary strategies in \overline{T} , i.e., for every $q \in \overline{Q}$, we have:*

$$val^{\overline{T}}(q) = \inf_{\mu \in \Xi_{Min}} \sup_{\chi \in \overline{\Sigma}_{Max}} \mathcal{A}_{Min}(q, \mu, \chi) = \sup_{\chi \in \Xi_{Max}} \inf_{\mu \in \overline{\Sigma}_{Min}} \mathcal{A}_{Max}(q, \mu, \chi) = val^{\widehat{T}}(q).$$

Proof. We have the following:

$$\begin{aligned} \inf_{\mu \in \Xi_{\text{Min}}} \sup_{\chi \in \overline{\Sigma}_{\text{Max}}} \mathcal{A}_{\text{Min}}(q, \mu, \chi) &= \inf_{\mu \in \Xi_{\text{Min}}} \sup_{\chi \in \Xi_{\text{Max}}} \mathcal{A}_{\text{Min}}(q, \mu, \chi) = \\ &= \sup_{\chi \in \Xi_{\text{Max}}} \inf_{\mu \in \Xi_{\text{Min}}} \mathcal{A}_{\text{Max}}(q, \mu, \chi) = \sup_{\chi \in \Xi_{\text{Max}}} \inf_{\mu \in \overline{\Sigma}_{\text{Min}}} \mathcal{A}_{\text{Max}}(q, \mu, \chi), \end{aligned}$$

where the first and last equalities follow from Lemma 3, and the second equality follows from Theorem 5.

Now we show that $\underline{\text{val}}^{\overline{T}}(q) \geq \underline{\text{val}}^{\widehat{T}}(q)$. The proof that $\overline{\text{val}}^{\overline{T}}(q) \leq \overline{\text{val}}^{\widehat{T}}(q)$ is similar and hence omitted.

$$\begin{aligned} \underline{\text{val}}^{\overline{T}}(q) &= \sup_{\chi \in \overline{\Sigma}_{\text{Max}}} \inf_{\mu \in \overline{\Sigma}_{\text{Min}}} \mathcal{A}_{\text{Max}}(q, \mu, \chi) \geq \sup_{\chi \in \Xi_{\text{Max}}} \inf_{\mu \in \overline{\Sigma}_{\text{Min}}} \mathcal{A}_{\text{Max}}(q, \mu, \chi) \\ &= \sup_{\chi \in \Xi_{\text{Max}}} \inf_{\mu \in \Xi_{\text{Min}}} \mathcal{A}_{\text{Max}}(q, \mu, \chi) = \underline{\text{val}}^{\widehat{T}}(q). \end{aligned}$$

The first inequality follows as $\Xi_{\text{Max}} \subseteq \overline{\Sigma}_{\text{Max}}$. The first equality holds by definition, the second equality is proved in the first paragraph of this proof, and the third equality follows from Theorem 5. From Lemma 4 we know that $\underline{\text{val}}^{\widehat{T}}(q) = \overline{\text{val}}^{\widehat{T}}(q)$. It follows that the average-time game on \overline{T} is determined, and there are optimal type-preserving boundary strategies in \overline{T} . \square

6.3 Determinacy of Average-Time Games on the Region Graph

Lemma 4. *If the strategies $\mu^* \in \Xi_{\text{Min}}$ and $\chi^* \in \Xi_{\text{Max}}$ are optimal for respective players in \overline{T} then for every $\varepsilon > 0$, we have that*

$$\sup_{\chi \in \overline{\Sigma}_{\text{Max}}} \mathcal{A}_{\text{Min}}(q, \mu_\varepsilon^*, \chi) \leq \underline{\text{val}}^{\overline{T}}(q) + \varepsilon \quad \text{and} \quad \inf_{\mu \in \overline{\Sigma}_{\text{Min}}} \mathcal{A}_{\text{Max}}(q, \mu, \chi_\varepsilon^*) \geq \underline{\text{val}}^{\overline{T}}(q) - \varepsilon,$$

for all $\mu_\varepsilon^* \in \widetilde{\Sigma}_{\text{Min}}^{(\mu^*, \varepsilon)}$ and $\chi_\varepsilon^* \in \widetilde{\Sigma}_{\text{Max}}^{(\chi^*, \varepsilon)}$.

Proof. Let $\mu^* \in \Xi_{\text{Min}}$ and $\chi^* \in \Xi_{\text{Max}}$ are optimal for respective players in \overline{T} . For all $\chi \in \overline{\Sigma}_{\text{Max}}$, $\varepsilon > 0$, and $\mu_\varepsilon^* \in \widetilde{\Sigma}_{\text{Min}}^{(\mu^*, \varepsilon)}$, we have the following:

$$\mathcal{A}_{\text{Min}}(q, \mu_\varepsilon^*, \chi) \leq \mathcal{A}_{\text{Min}}(q, \mu^*, \chi^{(q, \mu_\varepsilon^*)}) + \varepsilon \leq \mathcal{A}_{\text{Min}}(q, \mu^*, \chi^*) + \varepsilon = \underline{\text{val}}^{\overline{T}}(q) + \varepsilon.$$

The first inequality is by Corollary 2. The second inequality holds because χ^* is an optimal strategy and the equality is due to the fact that μ^* and χ^* are optimal. \square

Theorem 7. *The average-time game on \widetilde{T} is determined, and for every $q \in \overline{Q}$, we have $\underline{\text{val}}^{\widetilde{T}}(q) = \underline{\text{val}}^{\overline{T}}(q)$.*

Proof. Let $\mu^* \in \Xi_{\text{Min}}$ be an optimal strategy of player Min in $\overline{\mathcal{T}}$. Let us fix an $\varepsilon > 0$ and $\mu_\varepsilon^* \in \tilde{\Sigma}_{\text{Min}}^{(\mu^*, \varepsilon)}$.

$$\begin{aligned} \overline{\text{val}}^{\tilde{\mathcal{T}}}(q) &= \inf_{\mu \in \tilde{\Sigma}_{\text{Min}}} \sup_{\chi \in \tilde{\Sigma}_{\text{Max}}} \mathcal{A}_{\text{Min}}(q, \mu, \chi) \leq \sup_{\chi \in \tilde{\Sigma}_{\text{Max}}} \mathcal{A}_{\text{Min}}(q, \mu_\varepsilon^*, \chi) \leq \\ &\sup_{\chi \in \overline{\Sigma}_{\text{Max}}} \mathcal{A}_{\text{Min}}(q, \mu_\varepsilon^*, \chi) \leq \text{val}^{\overline{\mathcal{T}}}(q) + \varepsilon. \end{aligned}$$

The second inequality follows because $\mu_\varepsilon^* \in \tilde{\Sigma}_{\text{Min}}$ and the third inequality follows as $\tilde{\Sigma}_{\text{Max}} \subseteq \overline{\Sigma}_{\text{Max}}$. The last inequality follows from Lemma 4 because $\mu^* \in \Xi_{\text{Min}}$ is an optimal strategy in $\overline{\mathcal{T}}$. Similarly we show that for every $\varepsilon > 0$ we have that $\underline{\text{val}}^{\tilde{\mathcal{T}}}(q) \geq \text{val}^{\overline{\mathcal{T}}}(q) - \varepsilon$. Hence it follows that $\text{val}^{\tilde{\mathcal{T}}}(q)$ exists and its value is equal to $\text{val}^{\overline{\mathcal{T}}}(q)$. \square

7 Complexity

The main decision problem for average-time game is as follows: given an average-time game $\Gamma = (\mathcal{T}, L_{\text{Min}}, L_{\text{Max}})$, a state $s \in S$, and a number $B \in \mathbb{R}_\oplus$, decide whether $\text{val}(s) \leq B$.

Theorem 8. *Average-time games are EXPTIME-complete on timed automata with at least two clocks.*

Proof. From Theorem 3 we know that in order to solve an average-time game starting from an initial state of a timed automaton, it is sufficient to solve the average-time game on the set of states of the boundary region graph of the automaton that are reachable from the initial state. Observe that every region, and hence also every configuration of the game, can be represented in space polynomial in the size of the encoding of the timed automaton and of the encoding of the initial state, and that every move of the game can be simulated in polynomial time. Therefore, the value of the game can be computed by a straightforward alternating PSPACE algorithm, and hence the problem is in EXPTIME because $\text{APSPACE} = \text{EXPTIME}$.

In order to prove EXPTIME-hardness of solving average-time games on timed automata with two clocks, we reduce the EXPTIME-complete problem of solving countdown games [10] to it. Let $G = (N, M, \pi, n_0, B_0)$ be a countdown game, where N is a finite set of nodes, $M \subseteq N \times N$ is a set of moves, $\pi : M \rightarrow \mathbb{N}_+$ assigns a positive integer number to every move, and $(n_0, B_0) \in N \times \mathbb{N}_+$ is the initial configuration.

W.l.o.g we assume that there is an integer W such that $\pi(n_1, n_2) \geq W$ for every move $(n_1, n_2) \in M$. $(n, B) \in N \times \mathbb{N}_+$, first player 1 chooses a number $p \in \mathbb{N}_+$, such that $p \leq B$ and $\pi(n, n') = p$ for some move $(n, n') \in M$, and then player 2 chooses a move $(n, n'') \in M$, such that $\pi(n, n'') = p$; the new configuration is then $(n'', B - p)$. Player 1 wins a play of the game when a configuration $(n, 0)$ is reached, and he loses (i.e., player 2 wins) when a configuration (n, B)

is reached in which player 1 is stuck, i.e., for all moves $(n, n') \in M$, we have $\pi(n, n') > B$.

We define the timed automaton $\mathcal{T}_G = (L, C, S, A, E, \delta, \xi, F)$ by setting $C = \{b, c\}$; $S = L \times ((B_0)_{\mathbb{R}})^2$; $A = \{*\} \cup P \cup M$, where $P = \pi(M)$, the image of the function $\pi : M \rightarrow \mathbb{N}_+$;

$$L = \{*\} \cup N \cup \{(n, p) : \text{there is } (n, n') \in M, \text{ s.t. } \pi(n, n') = p\};$$

$$E(a) = \begin{cases} \{(n, \nu) : n \in N \text{ and } \nu(b) = B_0\} & \text{if } a = *, \\ \{(*, \nu) : \nu(c) = W\} & \text{if } a = *, \\ \{(n, \nu) : \exists (n, n') \in M, \text{ s.t. } \pi(n, n') = p \text{ and } \nu(c) = 0\} & \text{if } a = p \in P, \\ \{((n, p), \nu) : \pi(n, n') = p \text{ and } \nu(c) = p\} & \text{if } a = (n, n') \in M, \end{cases}$$

$$\delta(\ell, a) = \begin{cases} * & \text{if } \ell = n \in N \text{ and } a = *, \\ * & \text{if } \ell = * \text{ and } a = *, \\ (n, p) & \text{if } \ell = n \in N \text{ and } a = p \in P, \\ n' & \text{if } \ell = (n, p) \in N \times P \text{ and } a = (n, n') \in M; \end{cases}$$

$\xi(a) = \{c\}$, for every $a \in A \setminus \{*\}$ and $\xi(*) = \{b, c\}$. Note that the timed automaton \mathcal{T}_G has only two clocks and that the clock b is reset only in the special location $*$.

Finally, we define the average-time game on timed game automaton $\Gamma_G = (\mathcal{T}_G, L_1, L_2)$ by setting $L_1 = N$ and $L_2 = L \setminus L_1$. It is routine to verify that value of the average-time game at the state $(n_0, (0, 0)) \in S$ is W in the average-time game on Γ_G if and only if player 1 has a winning strategy (from the initial configuration (n_0, B_0)) in the countdown game G . \square

Acknowledgments. This work was partially supported by the EPSRC grants EP/E022030/1 and EP/F001096/1.

References

1. R. Alur, M. Bernadsky, and P. Madhusudan. Optimal reachability for weighted timed games. In *International Colloquium on Automata, Languages and Programming (ICALP)*, volume 3142 of *LNCS*, pages 122–133. Springer, 2004.
2. R. Alur and D. Dill. A theory of timed automata. *Theoretical Computer Science*, 126(2):183–235, 1994.
3. E. Asarin and O. Maler. As soon as possible: Time optimal control for timed automata. In F. W. Vaandrager and J. H. van Schuppen, editors, *International Workshop on Hybrid Systems: Computation and Control (HSCC)*, volume 1569 of *LNCS*, pages 19–30. Springer-Verlag, 1999.
4. P. Bouyer, T. Brihaye, V. Bruyère, and J. Raskin. On the optimal reachability problem on weighted timed automata. *Formal Methods in System Design*, 31(2):135–175, 2007.
5. P. Bouyer, E. Brinksma, and K. G. Larsen. Staying alive as cheaply as possible. In *International Workshop on Hybrid Systems: Computation and Control (HSCC)*, volume 2993 of *LNCS*, pages 203–218. Springer, 2004.

6. P. Bouyer, F. Cassez, E. Fleury, and K. G. Larsen. Optimal strategies in priced timed game automata. In *IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS)*, volume 3328 of *LNCS*, pages 148–160. Springer, 2004.
7. T. Brihaye, T. A. Henzinger, V. S. Prabhu, and J. Raskin. Minimum-time reachability in timed games. In *International Colloquium on Automata, Languages and Programming (ICALP)*, volume 4596 of *LNCS*, pages 825–837. Springer, 2007.
8. J. Filar and K. Vrieze. *Competitive Markov Decision Processes*. Springer, 1997.
9. D. Gillette. Stochastic games with zero stop probabilities. In *Contributions to the Theory of Games*, volume 39 of *Annals of Mathematics Studies*, pages 179–187. Princeton University Press, 1957.
10. M. Jurdziński, F. Laroussinie, and J. Sproston. Model checking probabilistic timed automata with one or two clocks. In *International Conference on Tools and Algorithms for Construction and Analysis of Systems (TACAS)*, volume 4424 of *LNCS*, pages 170–184. Springer, 2007.
11. M. Jurdziński and A. Trivedi. Reachability-time games on timed automata. In L. Arge, C. Cachin, T. Jurdziński, and A. Tarlecki, editors, *International Colloquium on Automata, Languages and Programming (ICALP)*, volume 4596 of *LNCS*, pages 838–849. Springer, 2007.
12. M. Jurdziński and A. Trivedi. Concavely-priced timed automata. In F. Cassez and C. Jard, editors, *International Conference on Formal Modeling and Analysis of Timed Systems (FORMATS)*, volume 5215 of *LNCS*, pages 48–62. Springer, 2008.
13. T. Liggett and S. Lipman. Stochastic games with perfect information and time average payoff. *SIAM Review*, 11:604–607, 1969.
14. D. A. Martin. Borel determinacy. *Annals of Mathematics*, 102:363–371, 1975.
15. D. A. Martin. The determinacy of Blackwell games. *Journal of Symbolic Logic*, 63(4):1565–1581, 1998.
16. M. L. Puterman. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. Wiley, 1994.
17. A. Trivedi. *Competitive Optimisation on Timed Automata*. PhD thesis, Department of Computer Science, The University of Warwick, 2009.
18. U. Zwick and M. Paterson. The complexity of mean payoff games on graphs. *Theoretical Computer Science*, 158:343–359, 1996.

A Proof of Proposition 4

In order to prove this proposition, we need the following result.

Proposition 9 ([11,17]). *Let $\alpha \in \mathbb{A}$ and regions $R, R', R'' \in \mathcal{R}$ be such that $R \xrightarrow{R''}_\alpha R'$. If $F : \Omega(R') \rightarrow \mathbb{R}$ is simple then $F_{(\alpha, R'')}^\oplus : \Omega(R) \rightarrow \mathbb{R}$, defined as $(s, R) \mapsto t(s, \alpha) + F(\text{succ}(q, (\alpha, R'')))$, is simple.*

Proof (Proof of Proposition 4). Let $\mu \in \Xi_{\text{Min}}$ and $\chi \in \Xi_{\text{Max}}$. We prove this lemma by induction on the value of n . The base case for $n = 0$ is trivial. Assume that for every $\mu \in \Xi_{\text{Min}}$ and $\chi \in \Xi_{\text{Max}}$ the function $\text{time}(\text{run}_k(\cdot, \mu, \chi)) : \overline{Q} \rightarrow \mathbb{R}_\oplus$ is regionally simple. To prove this proposition we now need to show that for $\mu \in \Xi_{\text{Min}}$ and $\chi \in \Xi_{\text{Max}}$ the function $\text{time}(\text{run}_{k+1}(\cdot, \mu, \chi))$ is regionally simple.

Let the strategies $\mu' \in \Xi_{\text{Min}}$ and $\chi' \in \Xi_{\text{Max}}$ be such that for every $q \in \overline{Q}$ the run $\text{run}_k(\text{succ}(q, \mu, \chi), \mu', \chi')$ be the length k suffix of $\text{run}_{k+1}(q, \mu, \chi)$. From inductive hypothesis we have that $\text{run}_k(\cdot, \mu', \chi')$ is regionally simple. Assume that $R \in \mathcal{R}_{\text{Min}}$ and let $\widehat{\mu}(\langle q \rangle) = (\alpha, R'')$ for every $q \in \overline{Q}(R)$. The treatment for the case where $R \in \mathcal{R}_{\text{Max}}$ is similar. Now for every $q = (s, R) \in \overline{Q}(R)$ we have that $\text{time}(\text{run}_{k+1}(q, \mu, \chi)) = t(s, \alpha) + \text{run}_k(\text{succ}(q, (\alpha, R'')), \mu', \chi')$, which from Proposition 9 is a simple function. \square

B Proof of Proposition 5

Proof. Let $\mu \in \Xi_{\text{Min}}$, $\chi \in \Xi_{\text{Max}}$ and $q = (s, R), q' = (s', R) \in \overline{Q}(R)$. We have

$$\begin{aligned} & \mathcal{A}_{\text{Min}}(q, \mu, \chi) - \mathcal{A}_{\text{Min}}(q', \mu, \chi) \\ &= \liminf_{n \rightarrow \infty} (1/n) \cdot \text{time}(\text{run}_n(q, \mu, \chi)) - \liminf_{n \rightarrow \infty} (1/n) \cdot \text{time}(\text{run}_n(q', \mu, \chi)) \\ &= \liminf_{n \rightarrow \infty} (1/n) \cdot (b - s(c) - b + s'(c)) = \liminf_{n \rightarrow \infty} (1/n) \cdot (s'(c) - s(c)) = 0. \end{aligned}$$

The first equality is by definition, the second follows from Proposition 4, and the last two equalities are trivial. In a similar manner we show that $\mathcal{A}_{\text{Max}}(q, \mu, \chi) = \mathcal{A}_{\text{Max}}(q', \mu, \chi)$. \square

C Proof of Proposition 6

In order to prove this Proposition 6 and Proposition 7, we need the following result.

Proposition 10 ([11,17]). *Let $a \in A$ and regions $R, R', R'' \in \mathcal{R}$ be such that $R \rightarrow_* R'' \xrightarrow{a} R'$. If $F : \Omega(R') \rightarrow \mathbb{R}$ is simple then for every $q = (s, R) \in \Omega(R)$, function $F_{(q, R'', a)}^\oplus : I \rightarrow \mathbb{R}$, defined as $t \mapsto t + F(\text{succ}(q, (t, R'', a)))$, is continuous and nondecreasing, where $I = \{t \in \mathbb{R}_\oplus : (s + t) \in \text{clos}(R'')\}$.*

Proof (Proof of Proposition 6). The proof is by induction on n . The base case, when $n = 0$, is trivial. In the rest of the proof we show that for $\chi \in \Xi_{\text{Max}}$, $\mu \in \overline{\Sigma}_{\text{Min}}$, and a configuration $q = (s, R) \in \overline{Q}$, we have that $\text{time}(\text{run}_{k+1}(q, \mu, \chi)) \geq \text{time}(\text{run}_{k+1}(q, \mu^{(q, \chi)}, \chi))$ assuming that the proposition holds for $n = k$. The proof for the case where $q \in \overline{Q}_{\text{Max}}$ is trivial. In the rest of the proof we assume that $q \in \overline{Q}_{\text{Min}}$.

Let us fix $\chi \in \Xi_{\text{Max}}$ and $\mu \in \overline{\Sigma}_{\text{Min}}$. Let $\text{run}_{k+1}(q, \mu, \chi)$ and $\text{run}_{k+1}(q, \mu^{(q, \chi)}, \chi)$ be $\langle q_0, \tau_1, q_1, \dots, q_{k+1} \rangle$ and $\langle q'_0, \tau'_1, q'_1, \dots, q'_{k+1} \rangle$, respectively, where $q_0 = q'_0 = q$. Notice that by definition the run types of both runs are the same. Hence for every index $i \leq k+1$ we have $q_i = (s_i, R_i)$ and $q'_i = (s'_i, R_i)$, and for every index $i \leq k+1$ we have $\tau_i = (t_i, R'_i, a_i)$ and $\tau'_i = (t'_i, R'_i, a_i)$.

Let $X \in \Xi_{\text{Max}}$ and $M \in \overline{\Sigma}_{\text{Min}}$ be such that $\text{run}_k(q_1, M, X)$ be length k suffix of $\text{run}_{k+1}(q, \mu, \chi)$. Notice that we assume that X is type-preserving. It is easy to see that

$$\text{time}(\text{run}_{k+1}(q, \mu, \chi)) = t_1 + \text{time}(\text{run}_k(q_1, M, X)).$$

From inductive hypothesis, we get that

$$\text{time}(\text{run}_{k+1}(q, \mu, \chi)) \geq t_1 + \text{time}(\text{run}_k(q_1, M^{(q_1, X)}, X)). \quad (3)$$

Since the strategies $M^{(q_1, X)} \in \Xi_{\text{Min}}$ and $X \in \Xi_{\text{Max}}$ are type-preserving, from Proposition 4 we get that $\text{time}(\text{run}_k(\cdot, M^{(q_1, X)}, X))$ is regionally simple. Let us denote the restriction of this function on domain $\overline{Q}(R_1)$ by $\mathcal{F} : \overline{Q}(R_1) \rightarrow \mathbb{R}$. Let us define the partial function $\mathcal{F}^\oplus_{(q, R'_1, a)} : \mathbb{R}_\oplus \rightarrow \mathbb{R}$ as $t \mapsto t + \mathcal{F}(\text{succ}(q, (t, R'', a)))$, for all $t \in \mathbb{R}_\oplus$, such that $(s+t) \in \text{clos}(R'_1)$. The following inequality follows from (3):

$$\text{time}(\text{run}_{k+1}(q, \mu, \chi)) \geq t_1 + \mathcal{F}(q_1) \geq \inf_t \{ \mathcal{F}^\oplus_{(q, R'_1, a)}(t) : s+t \in \text{clos}(R'_1) \}.$$

Since $\mu^{(q, \chi)}$ is a type-preserving boundary strategy of player Min, from equation (1), we know that $t'_1 = \inf \{ t : s+t \in \text{clos}(R'_1) \}$. Moreover from Proposition 10 we have that $\mathcal{F}^\oplus_{(q, R'_1, a)}$ is continuous and nondecreasing on the domain $\{t \in \mathbb{R}_\oplus : (s+t) \in \text{clos}(R'_1)\}$. Hence $\mathcal{F}^\oplus_{(q, R'_1, a)}(t'_1) = \inf_t \{ \mathcal{F}^\oplus_{(q, R'_1, a)}(t) : s+t \in \text{clos}(R'_1) \}$. Combining these facts, we get the following inequalities:

$$\text{time}(\text{run}_{k+1}(q, \mu, \chi)) \geq \mathcal{F}^\oplus_{(q, R'_1, a)}(t'_1) = t'_1 + \text{time}(\text{run}_k(q'_1, M^{(q_1, X)}, X))$$

Since $\text{run}_k(q'_1, M^{(q_1, X)}, X)$ is length k suffix of $\text{run}_{k+1}(q, \mu^{(q, \chi)}, \chi)$, we get the desired inequality. \square

D Proof of Proposition 7

Proof. The proof is by induction on n . The base case, when $n = 0$, is trivial. In the rest of the proof we show that for $\chi \in \Xi_{\text{Max}}$, $\mu \in \overline{\Sigma}_{\text{Min}}$, $\varepsilon > 0$, $\chi_\varepsilon \in \overline{\Sigma}_{\text{Max}}^{(\chi, \varepsilon)}$, and a configuration $q = (s, R) \in \overline{Q}$, we have that $\text{time}(\text{run}_{k+1}(q, \mu, \chi_\varepsilon)) \geq \text{time}(\text{run}_{k+1}(q, \mu^{(q, \chi_\varepsilon)}, \chi)) - k \cdot \varepsilon$, assuming that the proposition holds for $n = k$.

Let us fix $\chi \in \Xi_{\text{Max}}$, $\mu \in \overline{\Sigma}_{\text{Min}}$, $\varepsilon > 0$, and $\chi_\varepsilon \in \widetilde{\Sigma}_{\text{Max}}^{(\chi, \varepsilon)}$. Let $\text{run}_{k+1}(q, \mu, \chi_\varepsilon)$ and $\text{run}_{k+1}(q, \mu^{(q, \chi_\varepsilon)}, \chi)$ be $\langle q_0, \tau_1, q_1, \dots, q_{k+1} \rangle$ and $\langle q'_0, \tau'_1, q'_1, \dots, q'_{k+1} \rangle$, respectively, where $q_0 = q'_0 = q$. Notice that by definition the run types of both runs are the same. Hence for every index $i \leq k+1$ we have $q_i = (s_i, R_i)$ and $q'_i = (s'_i, R_i)$, and for every index $i \leq k+1$ we have $\tau_i = (t_i, R'_i, a_i)$ and $\tau'_i = (t'_i, R'_i, a_i)$.

Let $X \in \Xi_{\text{Max}}$ and $M \in \overline{\Sigma}_{\text{Min}}$ be such that $\text{run}_k(q_1, M, X_\varepsilon)$ be length k suffix of $\text{run}_{k+1}(q, \mu, \chi_\varepsilon)$. Notice that we assume that X is type-preserving. It is easy to see that

$$\text{time}(\text{run}_{k+1}(q, \mu, \chi_\varepsilon)) = t_1 + \text{time}(\text{run}_k(q_1, M, X_\varepsilon)).$$

From inductive hypothesis, we get that

$$\text{time}(\text{run}_{k+1}(q, \mu, \chi_\varepsilon)) \geq t_1 + \text{time}(\text{run}_k(q_1, M^{(q_1, X_\varepsilon)}, X)) - k \cdot \varepsilon. \quad (4)$$

Since the strategies $M^{(q_1, X_\varepsilon)} \in \Xi_{\text{Min}}$ and $X \in \Xi_{\text{Max}}$ are type-preserving boundary strategies, from Proposition 4 we get that $\text{time}(\text{run}_k(\cdot, M^{(q_1, X_\varepsilon)}, X))$ is regionally simple. Let us denote the restriction of this function on domain $\overline{Q}(R_1)$ by $\mathcal{F} : \overline{Q}(R_1) \rightarrow \mathbb{R}$. Let us define the partial function $\mathcal{F}_{(q, R'_1, a)}^\oplus : \mathbb{R}_\oplus \rightarrow \mathbb{R}$ as $t \mapsto t + \mathcal{F}(\text{succ}(q, (t, R'', a)))$, for all $t \in \mathbb{R}_\oplus$, such that $(s+t) \in \text{clos}(R'_1)$. The following inequality follows from (4):

$$\text{time}(\text{run}_{k+1}(q, \mu, \chi_\varepsilon)) \geq t_1 + \mathcal{F}(q_1) - k \cdot \varepsilon \geq \inf_t \{ \mathcal{F}_{(q, R'_1, a)}^\oplus(t) : s+t \in \text{clos}(R'_1) \} - k \cdot \varepsilon.$$

We need to consider two cases: $q \in \overline{Q}_{\text{Min}}$ and $q \in \overline{Q}_{\text{Max}}$.

- Assume that $q \in \overline{Q}_{\text{Min}}$. Since $\mu^{(q, \chi_\varepsilon)}$ is a type-preserving boundary strategy of player Min, from equation (1), we know that $t'_1 = \inf\{t : s+t \in \text{clos}(R'_1)\}$. Moreover from Proposition 10 we have that $\mathcal{F}_{(q, R'_1, a)}^\oplus$ is continuous and non-decreasing on the domain $\{t \in \mathbb{R}_\oplus : (s+t) \in \text{clos}(R'_1)\}$. Hence $\mathcal{F}_{(q, R'_1, a)}^\oplus(t'_1) = \inf_t \{ \mathcal{F}_{(q, R'_1, a)}^\oplus(t) : s+t \in \text{clos}(R'_1) \}$. Combining these facts, we get the following inequalities:

$$\text{time}(\text{run}_{k+1}(q, \mu, \chi_\varepsilon)) \geq t'_1 + \text{time}(\text{run}_k(q'_1, M^{(q_1, X_\varepsilon)}, X)) - k \cdot \varepsilon.$$

Since $\text{run}_k(q'_1, M^{(q_1, X_\varepsilon)}, X)$ is length k suffix of $\text{run}_{k+1}(q, \mu^{(q, \chi_\varepsilon)}, \chi)$, we get the following inequality:

$$\begin{aligned} \text{time}(\text{run}_{k+1}(q, \mu, \chi_\varepsilon)) &\geq \text{time}(\text{run}_{k+1}(q, \mu^{(q_1, \chi_\varepsilon)}, \chi)) - k \cdot \varepsilon \\ &\geq \text{time}(\text{run}_{k+1}(q, \mu^{(q_1, \chi_\varepsilon)}, \chi)) - (k+1) \cdot \varepsilon, \end{aligned}$$

as required.

- Assume that $q \in \overline{Q}_{\text{Max}}$. So far we have shown that

$$\text{time}(\text{run}_{k+1}(q, \mu, \chi_\varepsilon)) \geq t_1 + \mathcal{F}(q_1) - k \cdot \varepsilon. \quad (5)$$

Since \mathcal{F} is a simple function let $\mathcal{F}((s_1, R_1)) = b - s_1(c)$ for all $(s_1, R_1) \in \overline{Q}(R_1)$. For all $t \in \mathbb{R}_\oplus$ such that $s+t \in R'_1$ we have the following observation.

$$t + \mathcal{F}(\text{succ}(s, (t, a_1))) = \begin{cases} t + b & \text{if } c \in \xi(a_1) \\ b - s(c) & \text{otherwise.} \end{cases} \quad (6)$$

By Definition 12 we know that $t_1 \geq t'_1 - \varepsilon$. Combining this with (6) we get that

$$t_1 + \mathcal{F}(q_1) \geq t'_1 + \mathcal{F}(q'_1) - \varepsilon.$$

We can then rewrite (5) as the following:

$$\text{time}(\text{run}_{k+1}(q, \mu, \chi_\varepsilon)) \geq t'_1 + \mathcal{F}(q'_1) - (k+1) \cdot \varepsilon.$$

The term $\mathcal{F}(q'_1)$ represents the sum of the times of $\text{run}_k(q'_1, M^{(q_1, X_\varepsilon)}, X)$. Since $\text{run}_k(q'_1, M^{(q_1, X_\varepsilon)}, X)$ is length k suffix of $\text{run}_{k+1}(q, \mu^{(q, \chi_\varepsilon)}, \chi)$, we get the inequality

$$\text{time}(\text{run}_{k+1}(q, \mu, \chi_\varepsilon)) \geq \text{time}(\text{run}_{k+1}(q, \mu^{(q, \chi_\varepsilon)}, \chi)) - (k+1) \cdot \varepsilon,$$

as required. □